Supporting Information: Time Series Modeling of Varying Complexity Captures Subdiffusive Solute Dynamics and Predicts Long Timescale Behavior in Nanoscale Pores.

Benjamin J. Coscia Michael R. Shirts

January 3, 2020

Contents

S1 Setup and analysis scripts S2

S2 Solute Equilibration S3

S3 Estimating the Hurst Parameter S4

S4 Simulating Fractional Lévy Motion S5

S4.1 Truncated Lévy stable hop distributions S5

S4.2 Achieving the right correlation structure S6

S5 Verifying Markovianity S7

S6 Derivation of Passage Time Distributions S10

S7 Solute hopping and trapping behavior S11

S8 Stationarity of Solute Trajectories S14

S9 Tabular Anomalous Diffusion Parameters S15

S10 Tabular MSDDM parameters S16

S11 Mean first passage time simulations S17

S1 Setup and analysis scripts

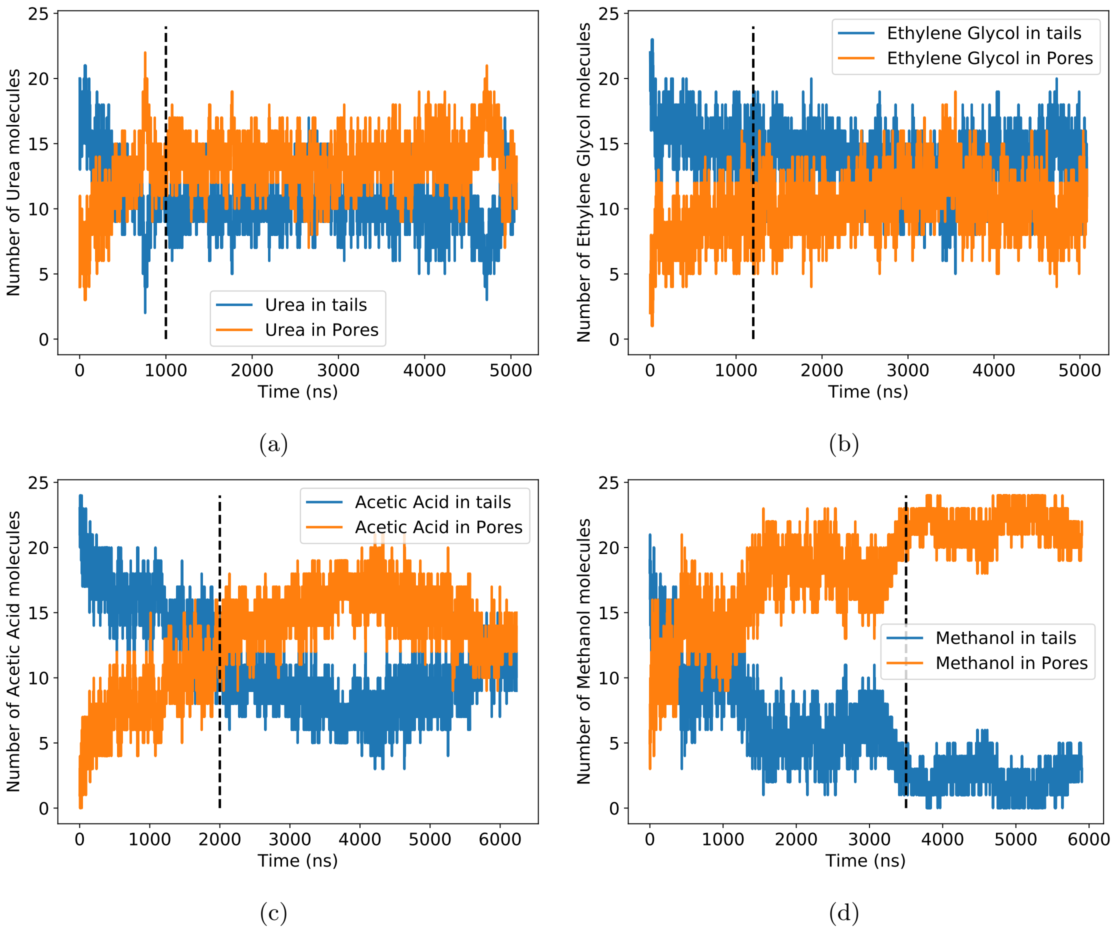
All python and bash scripts used to set up systems and conduct post-simulation trajectory analysis are available online at `https://github.com/shirtsgroup/LLC_Membranes`. Documentation for the `LLC_Membranes` repository is available at `https://llc-membranes.readthedocs.io/en/latest/`. Table S1 provides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

Script Name	Section	Description
`/setup/parameterize.py`	2.1	Parameterize liquid crystal monomers and solutes with GAFF
`/setup/build.py`	2.1	Build simulation unit cell
`/setup/place_solutes_pores.py`	2.1	Place equispaced solutes in the pore centers of a unit cell
`/setup/equil.py`	2.1	Equilibrate unit cell and run production simulation
`/analysis/solute_partitioning.py`	2.1	Determine time evolution of partition of solutes between pores and tails
`/timeseries/msd.py`	2.2	Calculate the mean squared displacement of solutes
`/analysis/sfbm_parameters.py`	2.2	Get subordinated fractional Brownian motion parameters by fitting to a solute's dwell and hop length distributions and positional autocorrelation function.
`/timeseries/ctrwsim.py`	2.2	Generate realizations of a continuous time random walk with the user's choice of dwell and hop distributions
`/timeseries/forecast_ctrw.py`	2.2	Combines classes from `sfbm_parameters.py` and `ctrwsim.py` to parameterize and predict MSD in one shot.
`/analysis/` `markov_state_dependent_dynamics.py`	2.3	Identify frame-by-frame state of each solute, construct a transition matrix and simulate realizations of the MSDDM model.
`/timeseries/mfpt_pore.py`	2.4	Simulate mean first passage time distributions using the AD or MSDDM model.

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 column 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 Solute Equilibration

We collected all data used for model generation after the solutes were equilibrated. We assumed a solute to be equilibrated when the partition of solutes in and out of the pore region stopped changing. The pore region is defined as within 0.75 nm of the pore center. We've plotted the partition versus time in Figure S1 and indicated the chosen equilibration time points.



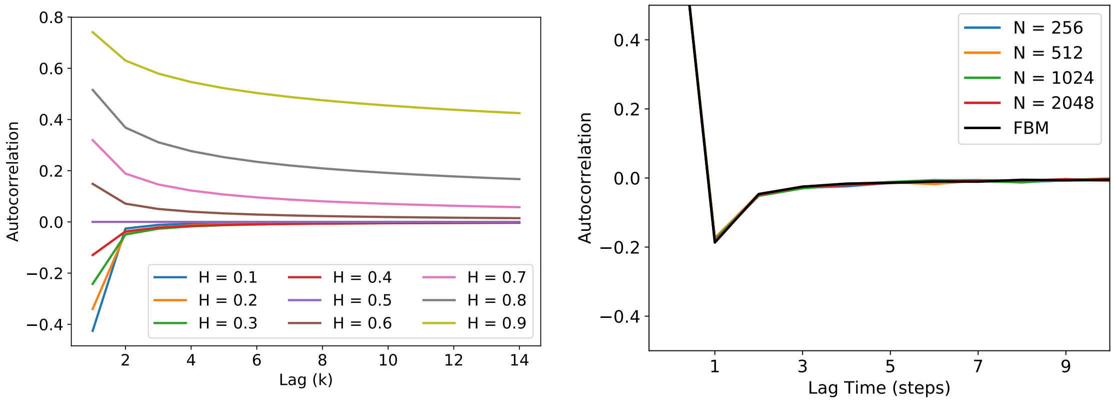
(a)

(b)

(c)

(d)

Figure S1: We considered a system to be equilibrated when the partition of solutes between the tails and pore plateaued. Our chosen equilibration point for each solute is indicated by the vertical black dashed line. (a) Urea equilibrates the fastest, after 1000 ns. (b) Ethylene glycol equilibrates after 1200 ns (c) The partition of acetic acid appears oscillate slowly. We considered it to be equilibrated after 2000 ns. (d) We considered methanol to be equilibrated after 3500 ns. Methanol nearly completely partitions into the tails.



(a)

(b)

Figure S2: (a) The analytical autocorrelation function of FBM decays to zero faster when H < 0.5 compared to when H > 0.5. (b) 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, for FLM, α=1.4.

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 6 in the main text):

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

In Figure S2a, we plotted Equation 1 for different values of H. When $H > 0.5$, Equation 1 decays slowly to zero meaning one needs to study large time lags with high frequency in order to accurately estimate H from the data. Fortunately, all of our solutes show anti-correlated motion, so most of the information in Equation 1 is contained within the first few lags.

The autocovariance function of fractional Lévy motion is different from fractional Brownian motion (see Equations 6 and 8 of the main text), but their autocorrelation structures are the same. The autocovariance 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 with the autocovariance function responding accordingly. 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 a well-defined structure and is independent of the coefficient in Equation 8 of the main text. In Figure S2b, 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. Therefore we are confident that we can use the same Hurst parameter as an input to both FBM and FLM simulations.

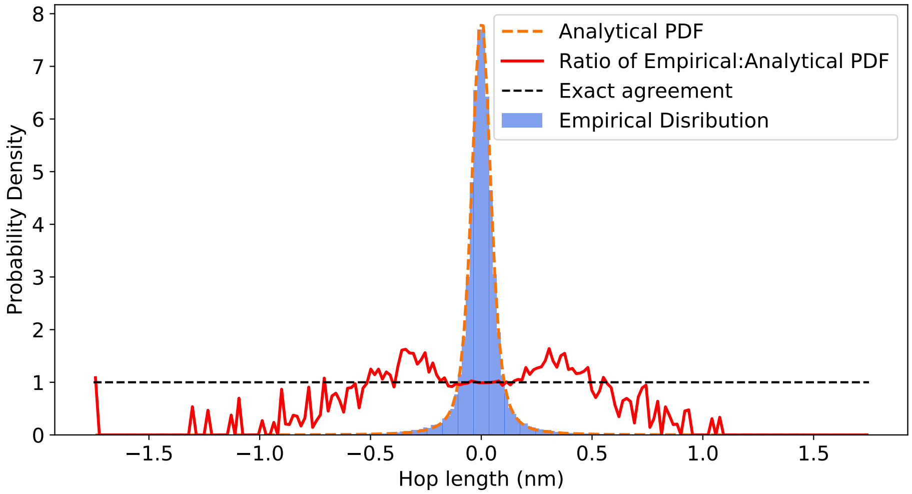


Figure S3: The ratio between the empirical and maximum likelihood theoretical distribution quantifies the quality of fit as function of hop length. The fit is near-perfect close to the mean. Intermediate hop lengths are over sampled, and the tails are under sampled. We used this type of plot to determine the appropriate place to truncate the Lévy stable distributions.

S4 Simulating Fractional Lévy Motion

S4.1 Truncated Lévy stable hop distributions

Determining where to truncate the hop distribution: A pure Lévy stable distribution has heavy tails which can lead to arbitrarily long hop lengths. Our distribution of hop lengths fits well to a Lévy distribution near the mean, but under samples the tails. In Figure S4 we compare the empirically measured transition emission distribution of the MSDDM for urea to its maximum likelihood fit to a Lévy stable distribution. The ratio between the two distributions at each bin is nearly 1 close to the center, indicating a near-perfect fit, larger than 1 slightly further from the center, suggesting that we slightly over sample intermediate hop lengths, and below 1 far from the center, indicating under sampling of extremely long hop lengths. Based on the plot, we chose a cut-off of 1 nm in order to compensate for over sampled intermediate hop lengths. We chose the same cut-off for all solutes.

Generating FLM realizations from a truncated Lévy distribution: 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. [1] 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.

We are unaware of a technique for simulating truncated FLM, therefore we devised our own based on the above discussion. 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. Therefore, we created a database meant 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 S4 shows the result of applying our correction. Note that generating this database requires a significant amount of simulation and still likely doesn't perfectly correct the truncation parameter. The output leads to a somewhat fuzzy, rather

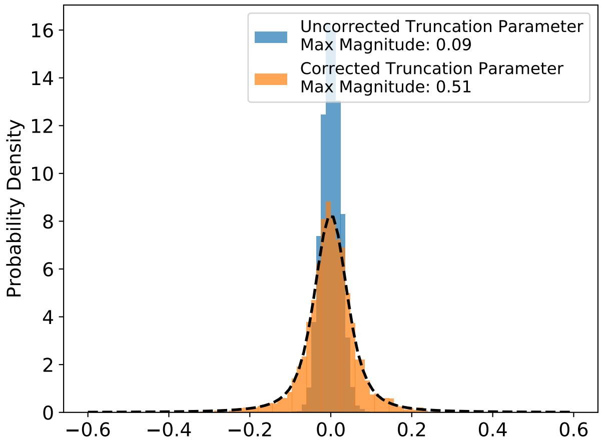


Figure S4: We can accurately truncate the marginal distribution of FLM innovations by applying a correction to the input truncation parameter. We generated FLM sequences and truncated the initial Lévy stable distribution (before Fourier transforming) at a value of 0.5. After correlation structure is added, the width of the distribution of fractional Lévy noise decreases significantly. We corrected the input truncation parameter with our database resulting in a distribution close to the theoretical distribution (black dashed line) with a maximum value close to 0.5.

than abrupt, cut-off of the output distribution. This is likely beneficial since we observe a small proportion of hops longer the chosen truncation cut-off. When the cut-off value is close to the Lévy stable σ parameter, as it is in our anomalous diffusion models, we observed that the tails of the truncated distribution tend to be undersampled. In order to maintain the distribution's approximate shape up to the cut-off value we recommend ensuring that the cut-off value is at least 2 times σ. However, this may lead to a slight over-prediction of the MSD.

S4.2 Achieving the right correlation structure

We simulated FLM using the algorithm of Stoev and Taqqu [1]. 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 the transition occurs at $H = 0.5$ 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 value 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 S5 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.

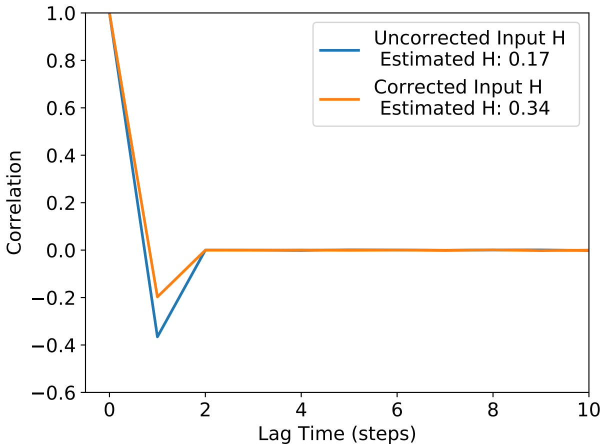


Figure S5: Correcting the Hurst parameter input to the algorithm of Stoev and Taqqu results in an FLM process with a more accurate correlation structure. We generated sequences with an input H of 0.35. We estimated H by fitting the autocorrelation function. Without the correction, H is underestimated, meaning realizations are more negatively correlated than they should be.

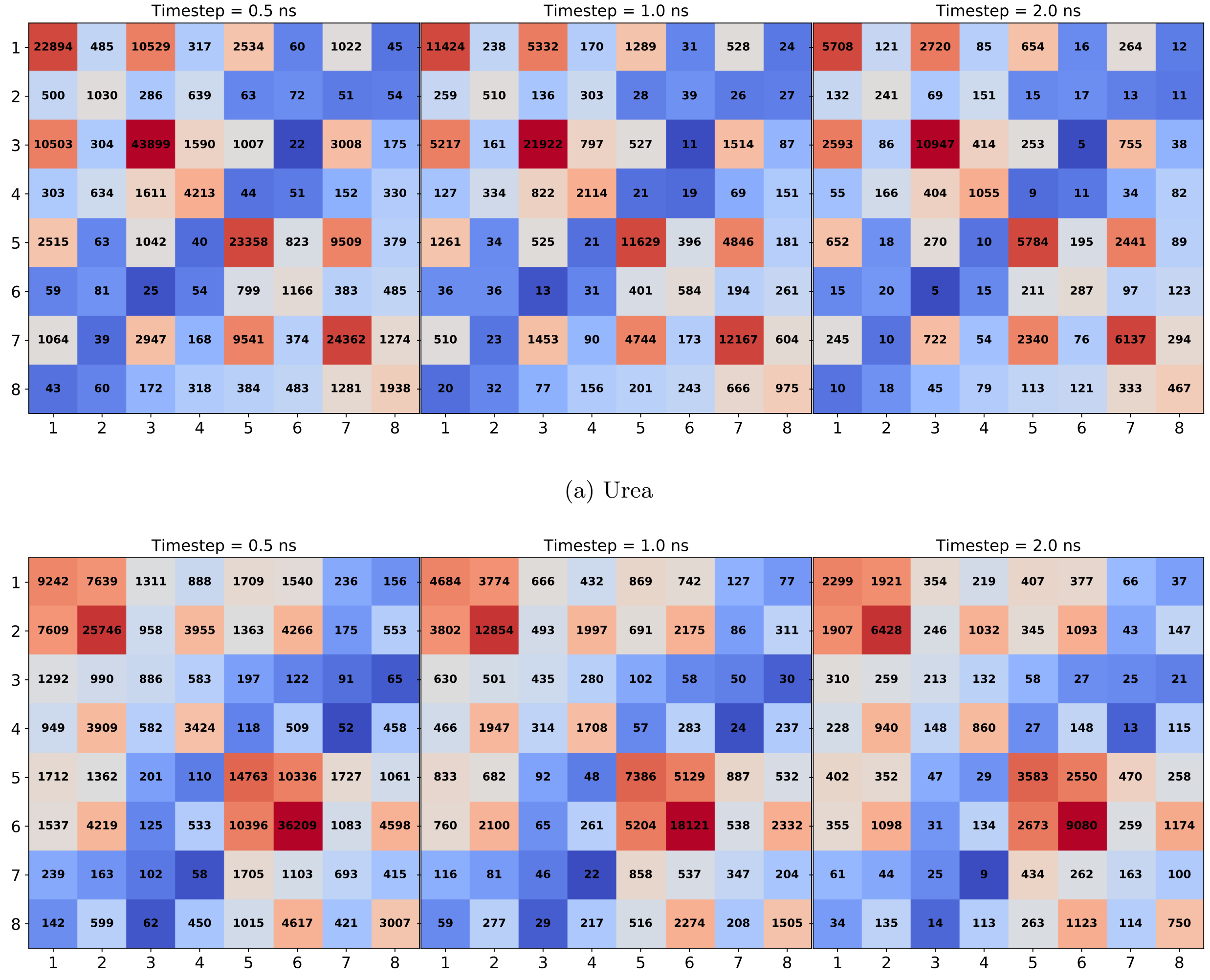
S5 Verifying Markovianity

We verified the Markovianity of our transition matrix, T, in two ways. First we ensured that the process satisfied detailed balance:

$$T_{i,j}P_i(t=\infty) = T_{j,i}P_j(t=\infty) \tag{2}$$

where P is the equilibrium distribution of states. This implies that the number of transitions from state i to j and from state j to i should be equal. Graphical representations of the count matrices show that this is true in Figure S6.

Second, we ensured that the transition matrix did not change on coarser time scales. In Figures S6 and S7, we show that increasing the length of time between samples does not change the properties of the count or probability transition matrices.



Timestep = 0.5 ns

	1	2	3	4	5	6	7	8
1	22894	485	10529	317	2534	60	1022	45
2	500	1030	286	639	63	72	51	54
3	10503	304	43899	1590	1007	22	3008	175
4	303	634	1611	4213	44	51	152	330
5	2515	63	1042	40	23358	823	9509	379
6	59	81	25	54	799	1166	383	485
7	1064	39	2947	168	9541	374	24362	1274
8	43	60	172	318	384	483	1281	1938

Timestep = 1.0 ns

	1	2	3	4	5	6	7	8
1	11424	238	5332	170	1289	31	528	24
2	259	510	136	303	28	39	26	27
3	5217	161	21922	797	527	11	1514	87
4	127	334	822	2114	21	19	69	151
5	1261	34	525	21	11629	396	4846	181
6	36	36	13	31	401	584	194	261
7	510	23	1453	90	4744	173	12167	604
8	20	32	77	156	201	243	666	975

Timestep = 2.0 ns

	1	2	3	4	5	6	7	8
1	5708	121	2720	85	654	16	264	12
2	132	241	69	151	15	17	13	11
3	2593	86	10947	414	253	5	755	38
4	55	166	404	1055	9	11	34	82
5	652	18	270	10	5784	195	2441	89
6	15	20	5	15	211	287	97	123
7	245	10	722	54	2340	76	6137	294
8	10	18	45	79	113	121	333	467

(a) Urea

Timestep = 0.5 ns

	1	2	3	4	5	6	7	8
1	9242	7639	1311	888	1709	1540	236	156
2	7609	25746	958	3955	1363	4266	175	553
3	1292	990	886	583	197	122	91	65
4	949	3909	582	3424	118	509	52	458
5	1712	1362	201	110	14763	10336	1727	1061
6	1537	4219	125	533	10396	36209	1083	4598
7	239	163	102	58	1705	1103	693	415
8	142	599	62	450	1015	4617	421	3007

Timestep = 1.0 ns

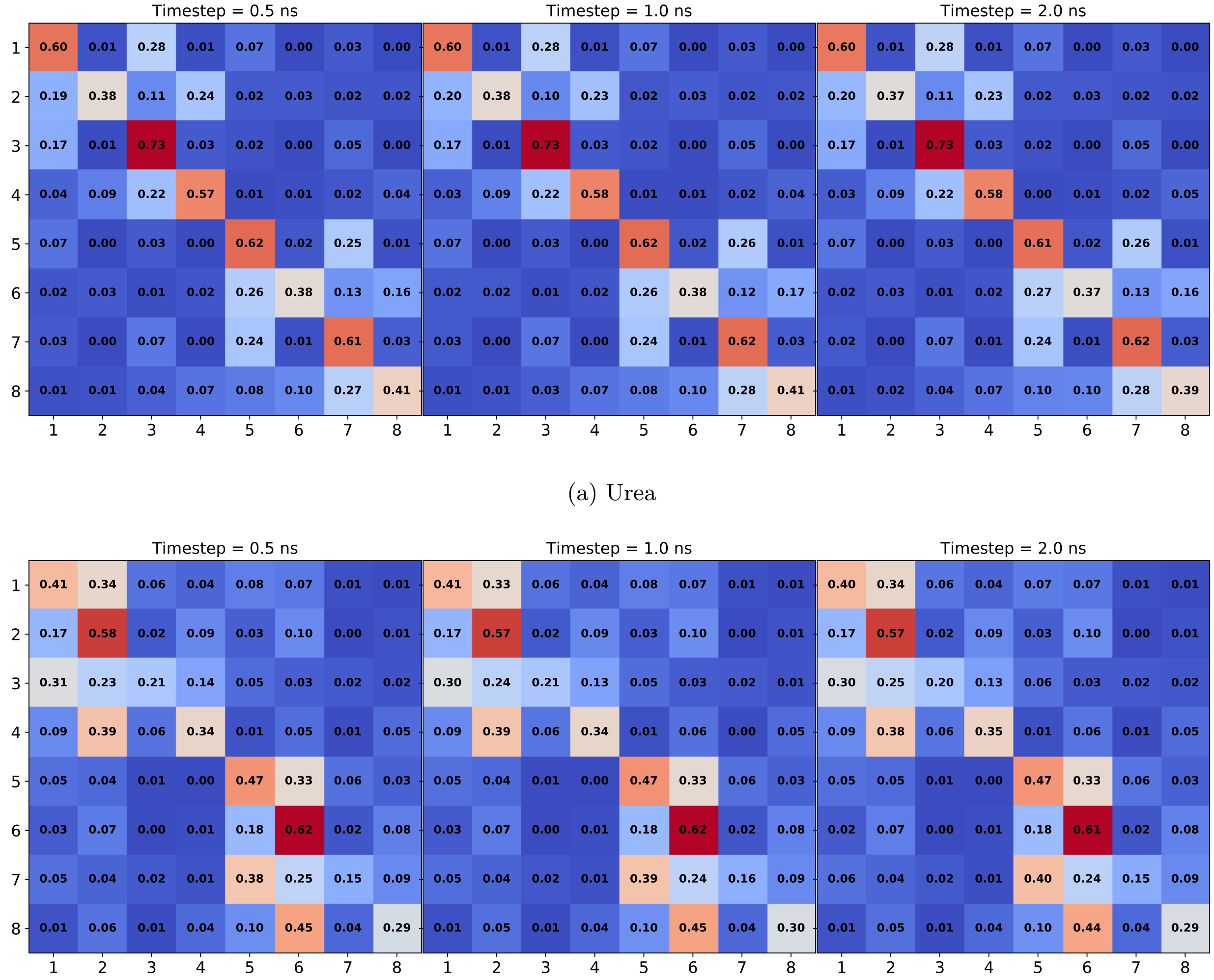
	1	2	3	4	5	6	7	8
1	4684	3774	666	432	869	742	127	77
2	3802	12854	493	1997	691	2175	86	311
3	630	501	435	280	102	58	50	30
4	466	1947	314	1708	57	283	24	237
5	833	682	92	48	7386	5129	887	532
6	760	2100	65	261	5204	18121	538	2332
7	116	81	46	22	858	537	347	204
8	59	277	29	217	516	2274	208	1505

Timestep = 2.0 ns

	1	2	3	4	5	6	7	8
1	2299	1921	354	219	407	377	66	37
2	1907	6428	246	1032	345	1093	43	147
3	310	259	213	132	58	27	25	21
4	228	940	148	860	27	148	13	115
5	402	352	47	29	3583	2550	470	258
6	355	1098	31	134	2673	9080	259	1174
7	61	44	25	9	434	262	163	100
8	34	135	14	113	263	1123	114	750

(b) Ethylene Glycol

Figure S6: The number of transitions from state i to j and j to i are very close indicating that our process obeys detailed balance. Detailed balance is conserved for different sized time steps.



(a) Urea

(b) Ethylene Glycol

Figure S7: As the timestep between observations increases, the probability transition matrix does not change signficantly.

S6 Derivation of Passage Time Distributions

To derive an analytical equation for the mean first passage time (Equation 11 of the main text), first consider an initial pulse spreading out over time with a fixed mean. We can solve for the time-dependent probability density of particle positions, p, by solving the one dimensional diffusion equation:

$$\frac{\partial p}{\partial t} = D\frac{\partial^2 p}{\partial z^2} \tag{3}$$

The appropriate initial and boundary conditions are:

$$BC1 : t > 0, z = \infty, p = 0$$

$$BC2 : t > 0, z = 0, \frac{\partial p}{\partial z} = 0$$

$$IC : t = 0, c = \delta(z)$$

It has been shown elsewhere that the solution to this equation is: [2]

$$p(z,t) = \frac{1}{\sqrt{4\pi Dt}} \exp\left(\frac{-z^2}{4Dt}\right) \tag{4}$$

We can make the substitution $z = z - vt$, where v represents a constant average velocity, in order to linearly shift the mean as a function of time:

$$p(z,t) = \frac{1}{\sqrt{4\pi Dt}} \exp\left(\frac{-(z-vt)^2}{4Dt}\right) \tag{5}$$

One can track the fraction of particles, F, that have crossed the pore boundary by integrating:

$$F(t) = \int_L^\infty p \; dz = \text{erfc}\left(\frac{L-vt}{2\sqrt{Dt}}\right) \tag{6}$$

where L is the pore length. This represents the cumulative first passage time distribution so we take its derivative in order to arrive at the first passage time distribution:

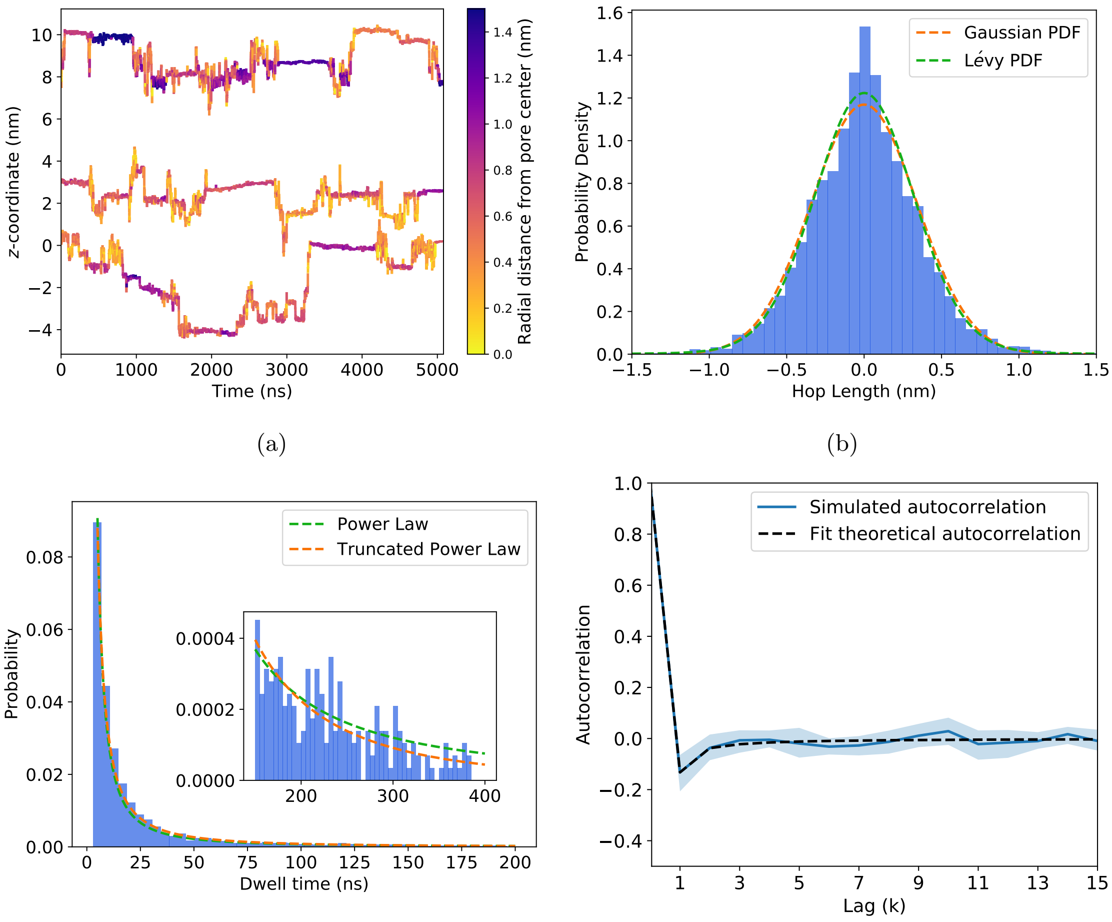
$$P(t) = -\frac{1}{\sqrt{\pi}} e^{-(L-vt)^2/(4Dt)} \left(-\frac{D(L-vt)}{4(Dt)^{3/2}} - \frac{v}{2\sqrt{Dt}}\right) \tag{7}$$

where the only free parameters for fitting are v and D. We calculated the expected value of Equation 7 in order to get the MFPT. Specifically, we used the python package `scipy.integrate.quad` to numerically integrate:

$$E[t] = \int_0^\infty tP(t)dt \tag{8}$$

S7 Solute hopping and trapping behavior

Analagous to Figure 2 of the main text, Figures S8–S10 demonstrate that all solutes exhibit the same kind of anti-correlated hopping and trapping behavior.



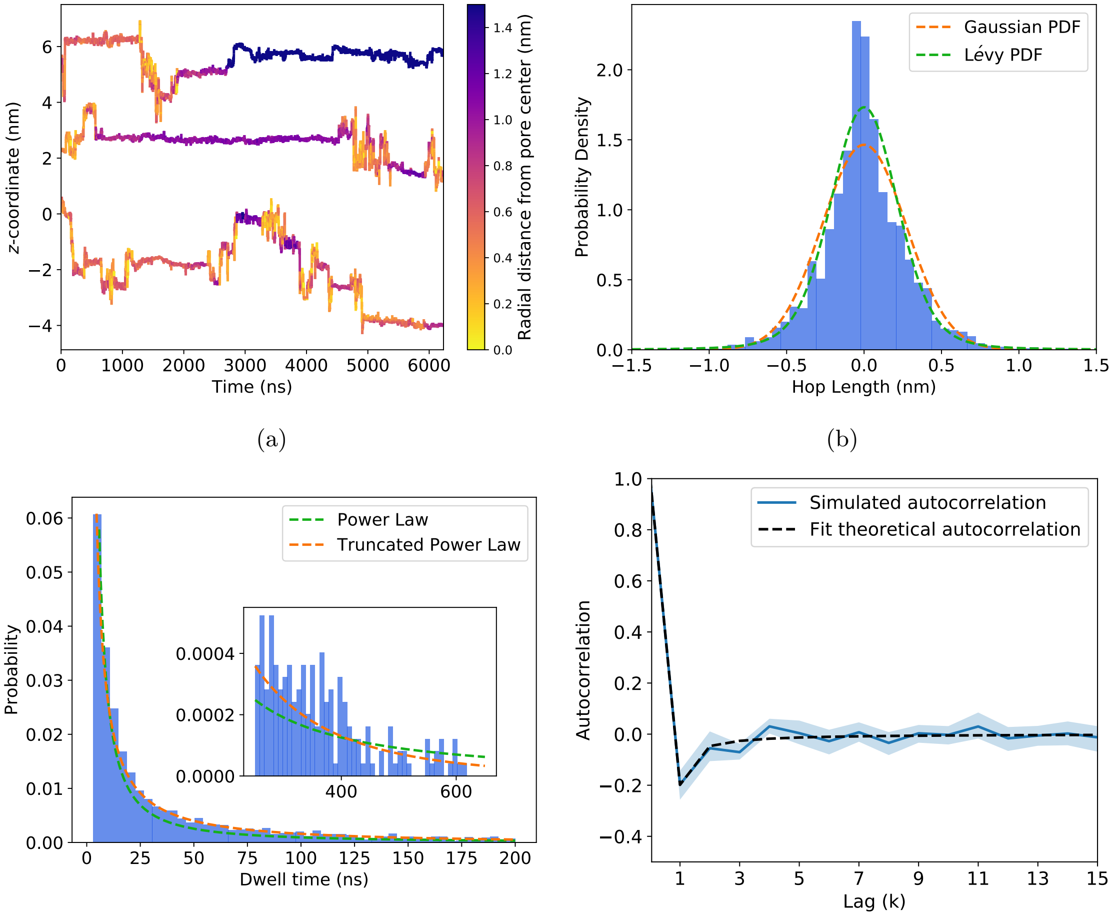
(a)

(b)

(c)

(d)

Figure S8: For ethylene glycol: (a) representative z-coordinate center of mass trajectories, (b) hop length distributions, (c) dwell time distributions and (d) hop autocorrelation functions. See Figure 2 and Section 3.1.1 in the main text for a more detailed discussion.



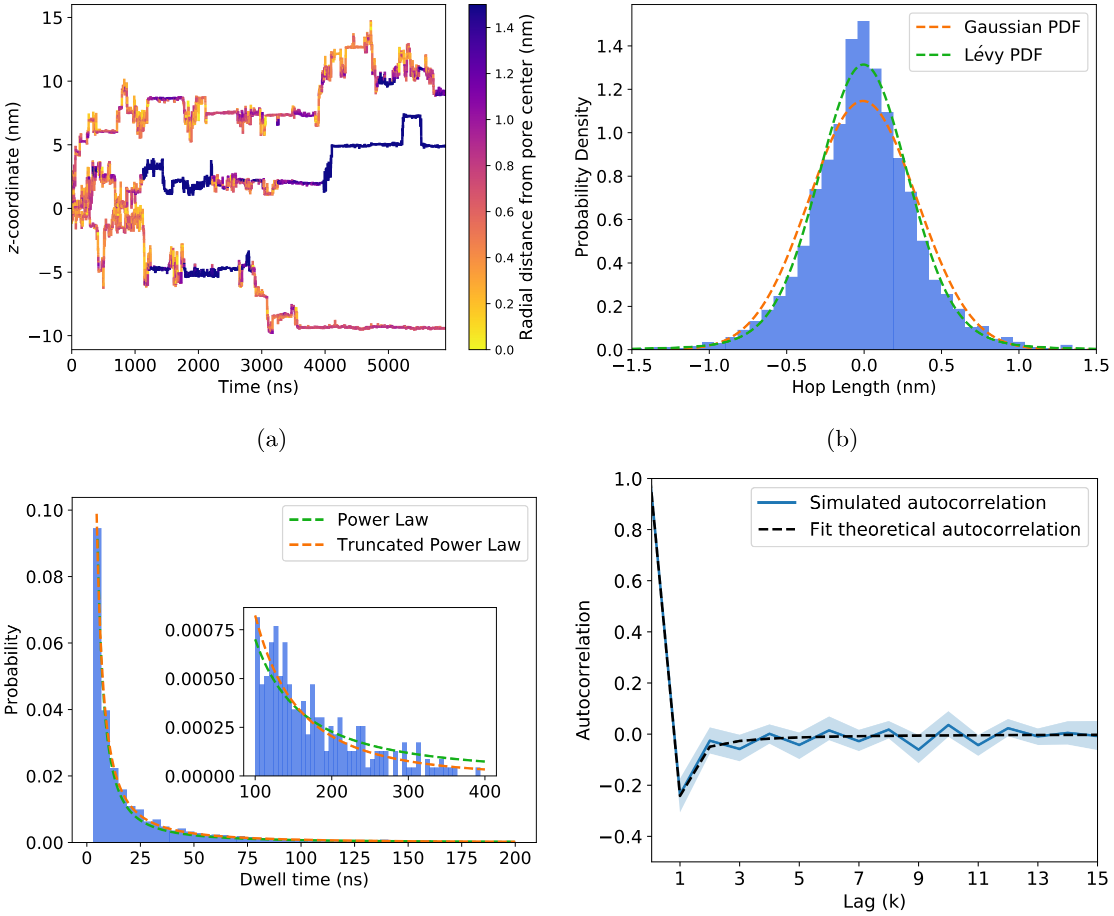
(a)

(b)

(c)

(d)

Figure S9: For acetic acid: (a) representative z-coordinate center of mass trajectories, (b) hop length distributions, (c) dwell time distributions and (d) hop autocorrelation functions. See Figure 2 and Section 3.1.1 in the main text for a more detailed discussion.



(a)

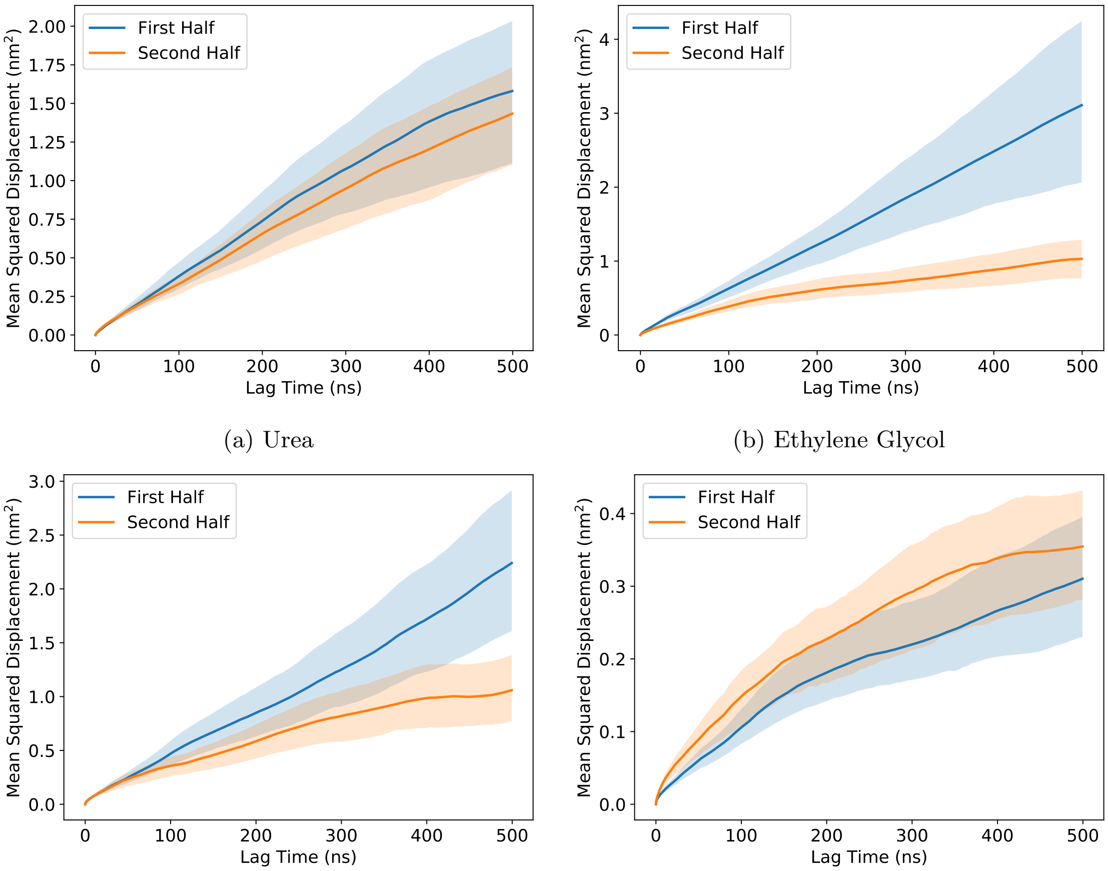
(b)

(c)

(d)

Figure S10: For methanol: (a) representative z-coordinate center of mass trajectories, (b) hop length distributions, (c) dwell time distributions and (d) hop autocorrelation functions. See Figure 2 and Section 3.1.1 in the main text for a more detailed discussion.

S8 Stationarity of Solute Trajectories



(a) Urea

(b) Ethylene Glycol

(c) Methanol

(d) Acetic Acid

Figure S11: The ensemble of solute trajectories may be stationary if the MSD calculated from different portions of the trajectory are the same. Here we plot the MSD calcluated up to a 500 ns time lag of the first and second halves of the equilibrated solute trajctories. Urea and acetic acid have similar MSDs, providing evidence of stationarity, while the MSDs of ethylene glycol and methanol are different suggesting that they are not.

S9 Tabular Anomalous Diffusion Parameters

The tables in the section are tabular representations of the parameters depicted in Figures 7 and 8 of the main text.

1 Mode Model	Parameters	Urea	Ethylene Glycol	Methanol	Acetic Acid
Dwell Distributions	$P(\alpha_d)$	0.57	0.62	0.62	0.45
	$P_T(\alpha_d, \lambda)$	0.40, 0.0024	0.47, 0.0030	0.44, 0.0040	0.08, 0.0033
Hop Distributions	$\mathcal{N}(\sigma)$	0.33	0.34	0.35	0.27
	$L(\sigma, \alpha_h)$	0.21, 1.84	0.23, 1.92	0.22, 1.80	0.16, 1.72
Correlation	$\gamma(H)$	0.37	0.40	0.30	0.34

Table S2: To create a 1 mode model for each solute, we parameterized a pure power law ($P(\alpha_d)$) and a truncated power law ($P_T(\alpha_d, \lambda)$) distribution to describe solute dwell times. Lower values of α_d lead to heavier power law tails and higher values of λ truncate the distribution at lower dwell times. We parameterized Gaussian ($\mathcal{N}(\sigma)$) and Lévy stable ($L(\sigma, \alpha_h)$) distributions to describe solute hop lengths. We assume the mean (μ) to be zero for these distributions and there to be no skewness ($\beta = 0$) in the Lévy stable distributions. High values of σ and lower values of α_h result in larger hops. Finally, we parameterized the hop autocorrelation function ($\gamma(H)$) to describe the degree of correlation between hops. Higher values of H display closer to Brownian behavior.

2 Mode Model						
	Parameters	Mode	Urea	Ethylene Glycol	Methanol	Acetic Acid
Dwell Distributions	$P(\alpha_d)$	1	0.69	0.69	0.90	0.58
		2	0.38	0.48	0.58	0.33
	$P_T(\alpha_d, \lambda)$	1	0.56, 0.0037	0.62, 0.0026	1.04, 0.0006	0.41, 0.0026
		2	0.00, 0.0027	0.06, 0.0049	0.30, 0.0054	0.00, 0.0021
Hop Distributions	$\mathcal{N}(\sigma)$	1	0.35	0.38	0.45	0.32
		2	0.24	0.23	0.32	0.17
	$L(\sigma, \alpha_h)$	1	0.24, 1.91	0.26, 1.99	0.31, 1.97	0.21, 1.91
		2	0.12, 1.50	0.15, 1.90	0.20, 1.85	0.09, 1.50
Correlation	$\gamma(H)$	–	0.37	0.40	0.30	0.34

Table S3: The two model parameterizes solute behavior in the pores and tails separately. Generally, movement is much more restricted in the tail region. To create a 2 mode model, we generated a set of parameters based on solute behavior as function of distance from the pore center. Mode 1 corresponds to solute behavior within 0.75 nm of the pore center and mode 2 corresponds to behavior greater than or equal to 0.75 nm from the pore center. Note that we used the same Hurst parameter for both modes due to a low number of sufficiently long sequences of hops in each mode. See Table S2 for descriptions of the parameters.

S10 Tabular MSDDM parameters

The following table is a tabular representation of the parameters depicted in Figure 12 of the main text.

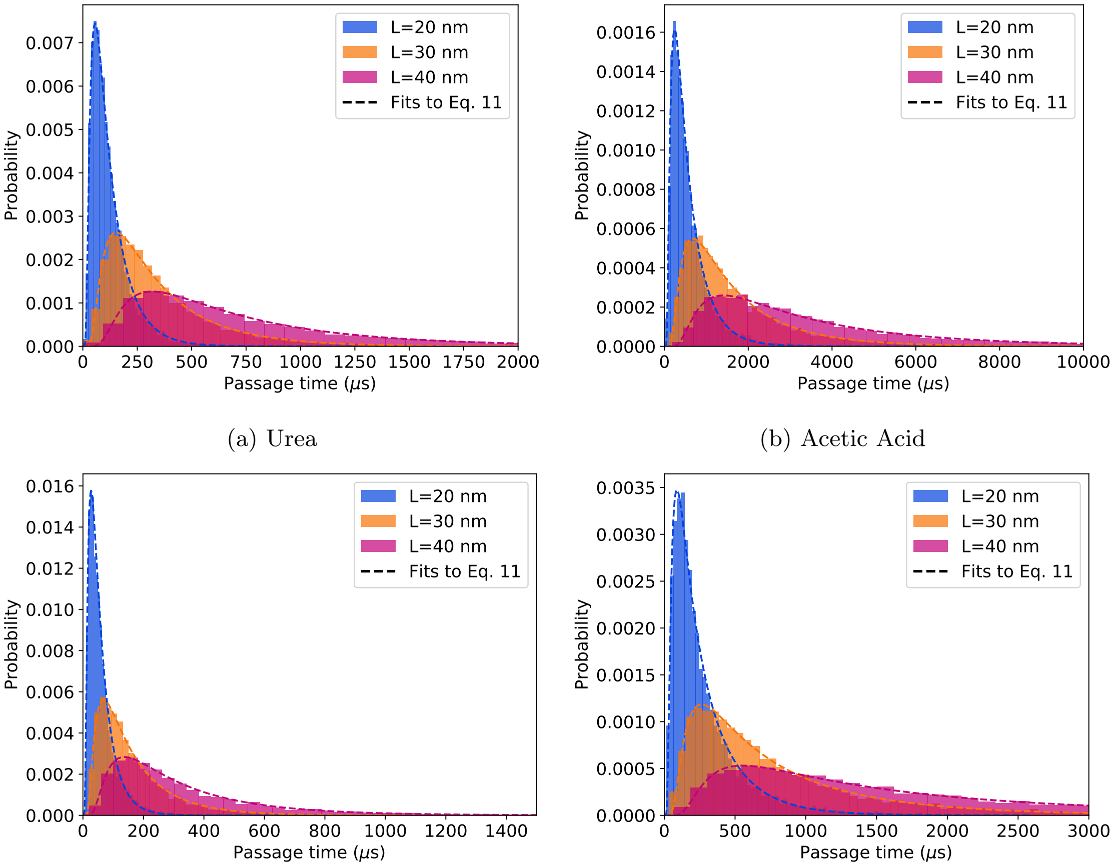
	Urea			Ethylene Glycol			Methanol			Acetic Acid		
State	H	α_h	σ	H	α_h	σ	H	α_h	σ	H	α_h	σ
1	0.10	1.79	0.034	0.09	1.68	0.045	0.11	1.56	0.052	0.10	1.78	0.035
2	0.06	1.80	0.033	0.09	1.75	0.037	0.07	1.63	0.043	0.08	1.88	0.032
3	0.11	1.88	0.030	0.11	1.86	0.030	0.02	1.80	0.036	0.04	2.00	0.030
4	0.10	1.95	0.027	0.04	1.91	0.028	0.02	1.75	0.036	0.04	2.00	0.027
5	0.19	1.34	0.048	0.15	1.40	0.062	0.10	1.28	0.074	0.13	1.47	0.048
6	0.15	1.45	0.040	0.11	1.52	0.040	0.03	1.50	0.042	0.09	1.70	0.038
7	0.15	1.61	0.032	0.05	1.60	0.040	0.28	1.20	0.043	0.08	1.77	0.031
8	0.11	1.71	0.028	0.05	1.74	0.030	0.04	1.83	0.037	0.01	2.00	0.030
T	0.34	1.42	0.036	0.37	1.44	0.045	0.35	1.45	0.057	0.34	1.54	0.040

Table S4: We calculated values of H, α_h and σ from MD simulation trajectories and used them to generate realizations of our MSDDM model. The states are defined in Table 2 of the main text except state T which describes the transition emissions.

S11 Mean first passage time simulations

Analytical fits to MFPT distributions

In Figures S12 and S13, we demonstrate the high quality of our analytical fits of Equation 7 to the distribution of solute first passage times derived from both the AD and MSDDM models. The histograms in Figure S13 are relatively noisy because we only generated 1000 realizations of the MSDDM versus 10000 of the AD model. In Figure S14, we justify our use of fewer MSDDM realizations by using subsets of the 10000 AD model realizations to show that one needs as few as 100 independent trajectory realizations at each pore length in order to reliably fit Equation 7 to the passage time distributions.



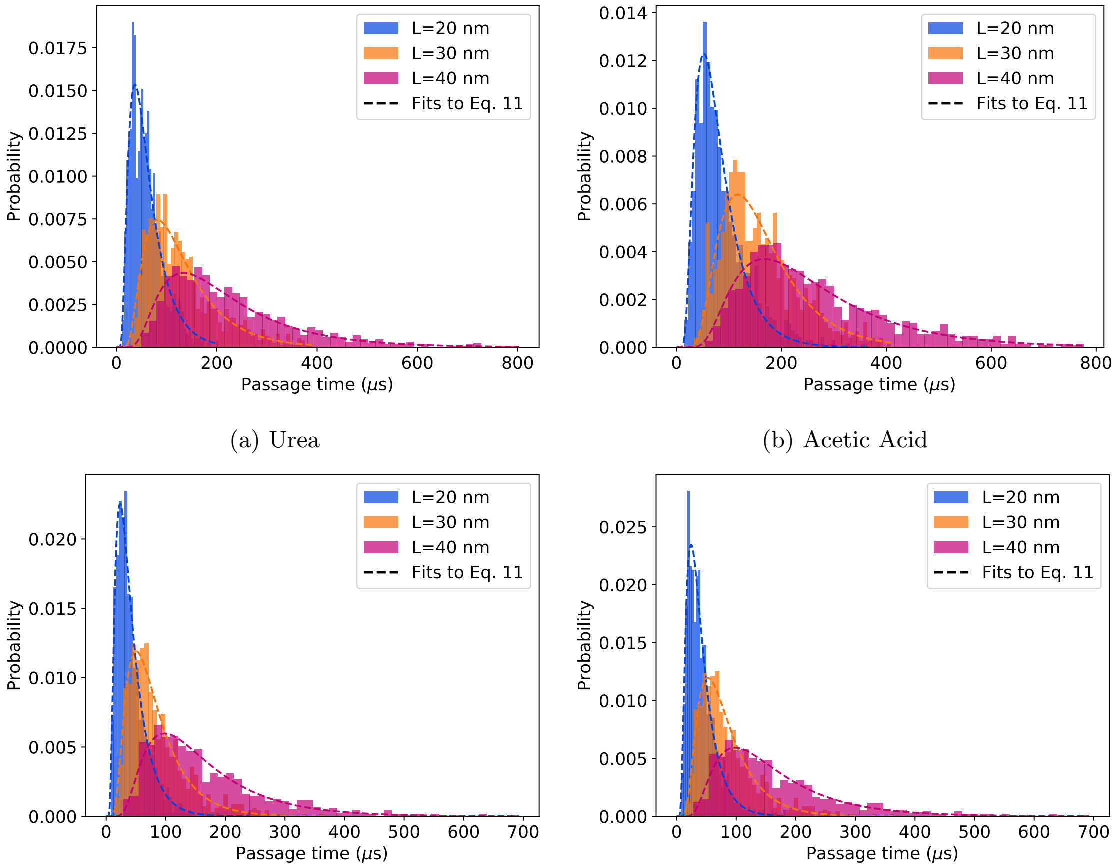
(a) Urea

(b) Acetic Acid

(c) Ethylene Glycol

(d) Methanol

Figure S12: We fit Equation 11 of the main text to the first passage time distributions generated by 10,000 realizations of the anomalous diffusion model.



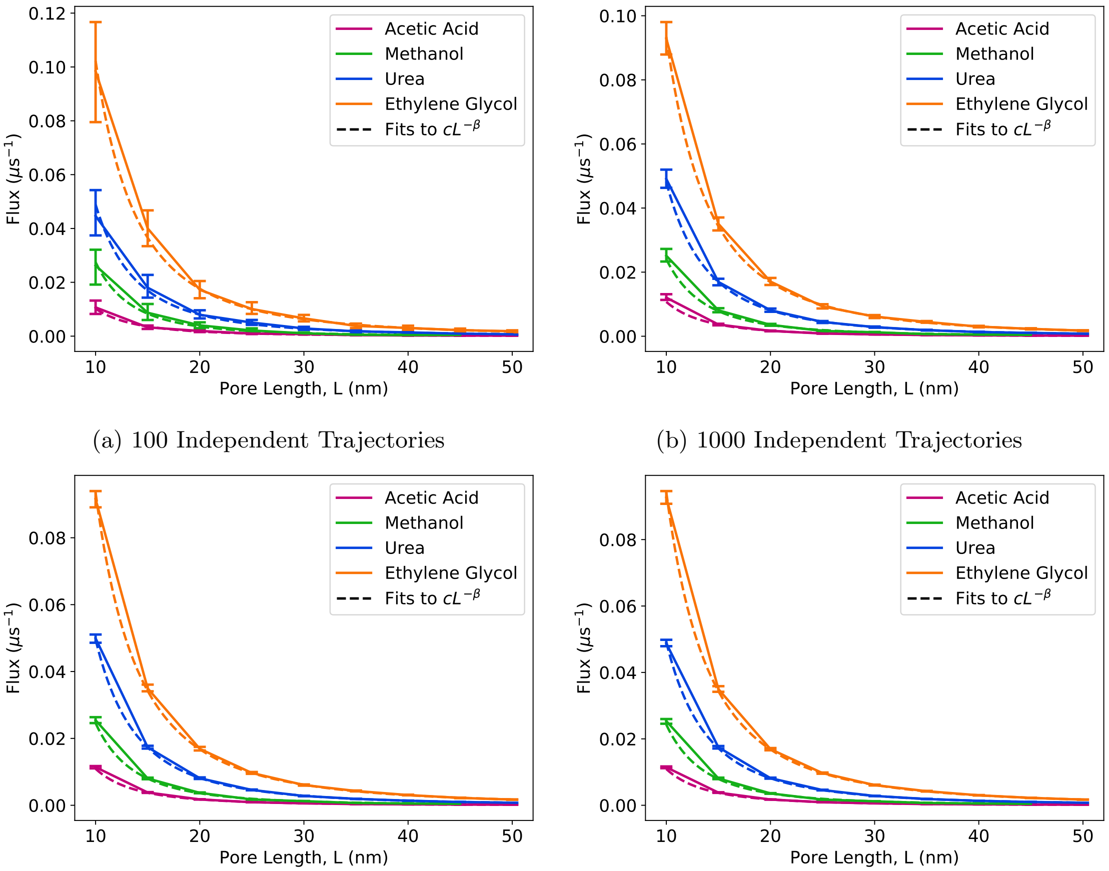
(a) Urea

(b) Acetic Acid

(c) Ethylene Glycol

(d) Methanol

Figure S13: We fit Equation 11 of the main text to the first passage time distributions generated by 1000 realizations of the Markov state dependent dynamical model. Note that we generated 10 times less realizations of the MSDDM which leads to noisier histograms. We show that this has a negligible effect on the fits in Figure S14.



(a) 100 Independent Trajectories

(b) 1000 Independent Trajectories

(c) 5000 Independent Trajectories

(d) 10000 Independent Trajectories

Figure S14: Even using a small number of independent trajectories, one can reliably calculate flux as a function of pore length. The uncertainty in the flux curves decreases as we add more independent trajectories.

References

[1] 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.

[2] E. L. Cussler, *Diffusion: Mass Transfer in Fluid Systems.* Cambridge University Press, 3 ed., 2009.