Pointer Algorithm User Manual

Introduction

The "pointer algorithm" is a mesoscopic simulation method that models the linear rheology of surfactant solutions containing wormlike micelles. It can be used for the following purposes:

1) to extract micelle parameters (e.g. micelle length, plateau modulus, and breakage time) from experimental small amplitude oscillatory shear rheology
2) to predict G' and G" rheology curves from a given set of micelle parameters
3) to compare the G' and G" rheology curves from specified micelle parameters to an experimental data set

The pointer algorithm is based on the Cates theory[1] that states that micelles, like entangled polymers, relax by diffusing through a tube formed by surrounding micelles, a process also known as reptation. However, unlike polymers, micelles can reversibly break and rejoin. In the pointer algorithm, reptation, breakage, and rejoining are simulated for an ensemble of micelles. High frequency relaxation modes, namely Rouse[2] and bending[3] modes, are also added analytically to the pointer algorithm, and the effects of chain length fluctuations and constraint release (double reptation) are considered in the simulation as well. A detailed explanation of the pointer algorithm can be found in [4].

Version history

The pointer algorithm was originally developed in the Larson Lab by Weizhong Zou, and has been modified by Grace Tan. It is written in Fortran (F90) and several notable versions are described below.

[unmerged] – This version of the pointer algorithm models the relaxation of well-entangled micelles. It was shown to be able to match the rheology of a couple common surfactant/salt systems from literature[4] and was fit to several SLE1S+CAPB/NaCl solutions[5,6]. A full description of this version is found in Ref. [4].

3.1 – In this version of the pointer algorithm, additional contributions to relaxation from unentangled micelles (shorter than the entanglement length) are added to the simulation. These added relaxation mechanisms, explained in Ref. [7], allow the pointer algorithm to be applied to surfactant solutions at lower concentrations than previously possible.

3.3 – After a comparison of the pointer algorithm with the more highly resolved slip-spring model[8], we found that the assumption that longitudinal (slow) Rouse modes can be neglected because entanglements impede relaxation along the micelle seems to be incorrect. Instead, for well-entangled micelles (an average of >15 entanglements per micelle), both fast and longitudinal Rouse modes must be considered, and if micelles are weakly entangled (an average

of <15 entanglements per micelle), an unfractionated, full spectrum of Rouse modes best describes the high-frequency data. These additional Rouse modes were added to the pointer algorithm in this version, as described in Ref. [9], which also shows improved fits to SLE1S+ CAPB/NaCl rheological data.

Preparing the simulation input file

To run a pointer algorithm simulation, there is a single input file, titled INPUT_[version].DAT. The instructions shown here will be for version 3.3 and are mostly, but not exactly, applicable to earlier versions. Figure 1 below shows a sample input file for reference. In the sections below, inputs are references as [line number].[column number] and "yes" and "no" are designated as "Y" and "N" respectively. This section provides general instructions for creating an input file; examples with sample numerical calculations and results are given below in the "Example simulations" section.

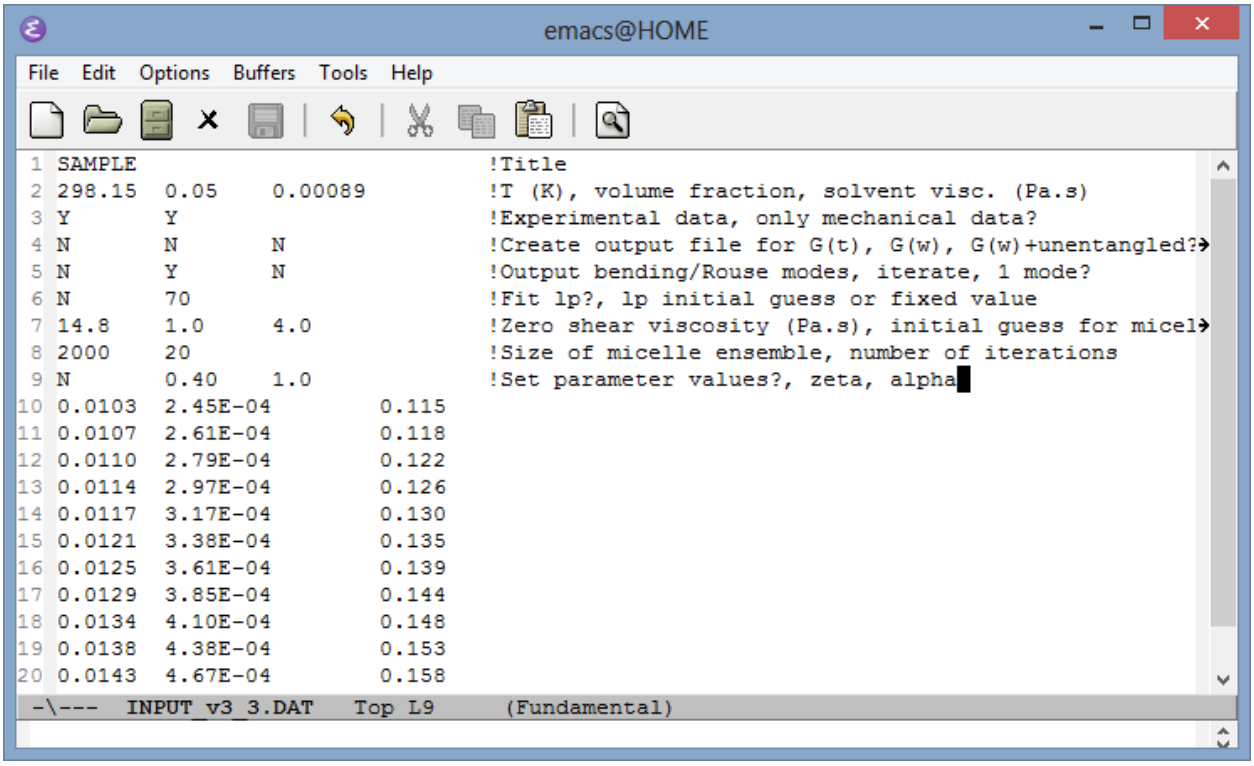

Figure 1: Sample input file for v3.3 of the pointer algorithm

1. Title – This can be the sample ID, a description of the surfactant/salt and concentration, or any other kind of identifier. It has no effect on how the simulation runs.

2.1. Temperature [K] – The temperature at which the experimental data were collected or at which you want to generate predictive rheology curves.

2.2. Micelle volume fraction ϕ – Ratio of the micelle volume to the total solution volume. For most systems that we've investigated, the volume fraction is within 10% of the weight fraction, so weight fraction can be used if volume fraction cannot be calculated or estimated another way.

2.3. Solvent viscosity η_s [Pa·s] – The solvent viscosity can either be that of the salt solution without surfactant, if measured separately, or if not, the viscosity of pure water at the appropriate temperature.

3.1. Are experimental data present? [Y/N]

- Y to run an iterative pointer algorithm simulation that fits the experimental data and extracts micelle parameters or a simulation that compares input micelle parameters to experimental data without iterating to fit the experimental rheology
- N to run a predictive pointer algorithm simulation for a given set of micelle parameters without comparing them to any experimental data

3.2. Are only mechanical data present? [Y/N]

- Y if the data are from a mechanical rheometer and go up to ~100-200 rad/s (also select yes if no data are present at all)
- N if there are high-frequency data (up to or greater than ~100,000 rad/s) from DWS or another experimental method

4.1. Should a file containing G(t) from reptation be generated? [Y/N] – If this option is selected, a file containing the G(t) curve from reptation and its best fit from the genetic algorithm (fitting with multiple Maxwell modes) for the current iteration will be outputted.

4.2. Should a file containing G'(ω) and G"(ω) from reptation be generated? [Y/N] – This option outputs a file containing the G(t) (reptation) curve for the current iteration transformed into the frequency domain.

4.3. Should the simulation output G'(ω) and G"(ω) with the contribution of unentangled micelles in a separate file? [Y/N] – This file contains G' and G" from reptation with relaxation from unentangled micelles (rotary relaxation, Rouse modes, and bending modes) added.

5.1. Should the simulation output the contributions to G'(ω) and G"(ω) from high frequency relaxation modes in separate files? [Y/N] – If this option is selected, the simulation will generate separate files with G' and G" for 1) rotary relaxation of unentangled micelles, 2) Rouse modes for unentangled micelles, 3) bending modes for unentangled micelles, 4) Rouse modes for entangled micelles, and 5) bending modes for entangled micelles.

5.2. Is an iterative simulation being run? [Y/N]

- Y if fitting to experimental data
- N if predicting the rheology of a specific set of micelle parameters (either independently of experimental data or to compare to experimental data)

5.3. Should the simulation try to fit G(t) from reptation with a single Maxwell mode? [Y/N] – The genetic algorithm that converts G(t) from the time to frequency domain allows a minimum of 2 Maxwell modes. With this option, the simulation will first try to fit G(t) with a single Maxwell mode. If the error is low enough, the simulation continues with the 1 Maxwell mode; if the error is too high it uses the genetic algorithm to find a better fit.

6.1. Is the persistence length a fitting parameter? [Y/N]
- Y if high frequency data are available and you want the simulation to extract the persistence length from the data
- N if no high frequency data are available or you want to set the persistence length yourself

6.2. Initial guess or fixed value for the persistence length l_p [nm] – For iterative simulations without high frequency data or predictive simulations, the persistence length is a required input parameter that would have to come from another experimental method or literature. For iterative simulations with high frequency data, the value for l_p entered here is either the starting value of l_p for the first iteration or chosen to be fixed depending on what was entered for input 6.1.

7.1. Zero shear viscosity η_0 [Pa·s] – If the zero shear viscosity was not measured for the solution of interest, it can be extracted from the slope of G" at low frequency. This value will not affect how the simulation converges, but the simulation may not think it's converged even if it has and keep running. The zero shear viscosity is not required as an input to run a predictive pointer algorithm simulation but can help in judging how well the predicted rheology curves match experimental data if making such a comparison.

7.2. Initial guess or value of the average micelle length $\langle L \rangle$ [μm] – For iterative fitting simulations, a better initial guess for $\langle L \rangle$ can decrease the number of iterations it takes before convergence. As a ballpark estimate, for solutions with η_0 < 10 Pa·s, $\langle L \rangle_0$ = 1 μm is a reasonable starting guess, increasing to $\langle L \rangle_0$ = 5-10 μm for 10 Pa·s < η_0 < 100 Pa·s.

7.3. Micelle diameter d [nm] – This input parameter cannot be determined from rheology. The literature value for a variety of systems is ~4 nm.

8.1. Number of micelles N in the simulated ensemble – The number of micelles in the ensemble needs to be large enough that the length distribution is not overly discretized, but more micelles take more time to simulation. From tests of the ensemble size, we have determined that 2000 micelles balances getting a good length distribution with simulation time. [Note that fewer micelles can be used, but certain parameters may be under or overestimated. If not running an iterative simulation and the micelles are very long the ensemble size can be decreased to ~500 micelles without overly affecting the predicted rheology curves to get the simulation to finish in a reasonable amount of time.]

8.2. Number of iterations – Our standard for an average set of experimental data is 20 iterations and 5 days of compute time on a high-performance computing cluster.

9.1. Is the simulation being started with all parameters specified? [Y/N] – Out of the five independent micelle parameters (d, l_p, $\langle L \rangle$, ζ, α), d cannot be determined from rheology and must be specified, l_p can be extracted from high-frequency data but at least requires a starting value, $\langle L \rangle$ needs a starting value as input, and ζ and α can both either start from user-input values or from simulation-estimated values.
- Y if running an iterative simulation and all parameters have been pre-calculated or restarting a simulation from a previous set of parameters. Also enter yes if running a predictive pointer algorithm simulation, for which all independent micelle parameters must be inputted. (See "Calculating Micelle Parameters" below for how to extract micelle parameters from experimental rheology using previously developed correlations.)
- N if running an iterative pointer algorithm simulation with no initial guesses for ζ and α

9.2. Initial guess or value for dimensionless breakage time ζ (not required if input 9.1 = N)

9.3. Initial guess or value for semi-flexibility factor α (not required if input 9.1 = N)

10-EOF. Experimental rheology data in three columns, in the order ω [rad/s], G'(ω) [Pa], G"(ω) [Pa]. See Figure 2 below about preparing experimental data for input. In particular, especially when using mechanical data only, watch for poor data at the lowest frequencies where the modulus may be lower than the physical limits of the rheometer and the highest frequencies where inertial effects can begin to affect the data.

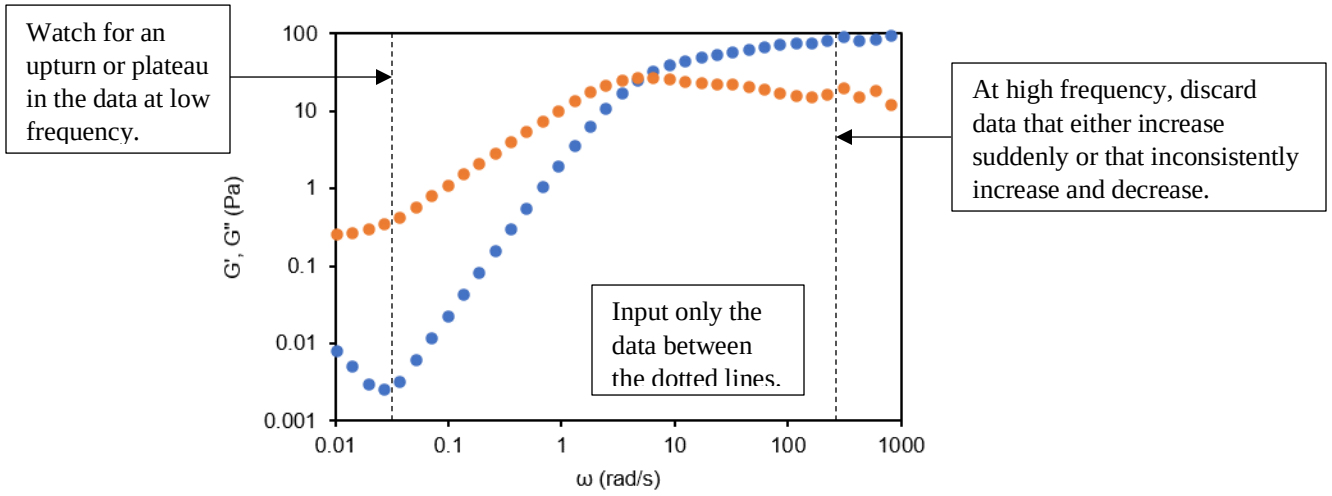

Figure 2: Example experimental rheology data

Calculating micelle parameters

The independent fitting parameters α, l_p, $\langle L \rangle$, and ζ can be estimated from experimental rheological data using correlations derived from pointer algorithm simulations. The process is summarized below.

1) Obtain solution parameters T (K), ϕ, and η_s (Pa·s).

2) Get experimental parameters G'_{min} (Pa), G''_{min} (Pa), and ω_{c1} (rad/s) from the rheology data. G'_{min} and G''_{min} are the values of G' and G" at the frequency where G" has a minimum. If there is no minimum in G", either the maximum of the ratio G'(ω)/G"(ω) or the limit of 1 can be used in the calculations below. ω_{c1} is the first crossover frequency, where the G' and G" intersect at low frequency.

3) Calculate G_0 (Pa) from the correlation

$\frac{G_0}{G'_{min}} = \frac{4.25}{G'_{min}/G''_{min}} + 0.625$ if $G'_{min}/G''_{min} < 10$ or take

$G_0 \approx G'_{min}$ if $G'_{min}/G''_{min} > 10$. (Eq. 1)

4.1) If l_p (m) is known or will be specified, calculate α from the crossover formula

$$G_0 = \frac{\alpha^3}{3+\alpha^3} 9.75 \frac{k_B T}{\alpha^3 l_p^{1.8}} + \frac{3}{3+\alpha^3} \frac{28}{5\pi} \frac{\phi k_B T}{d^2 l_e} \quad \text{(Eq. 2)}$$

4.2) To extract l_p from the high-frequency data, perform steps 4.1 and 4.2 simultaneously, solving the crossover formula while minimizing the error of fitting bending modes to the high-frequency data at the same time. Equation for bending modes:

$$G'' - \omega\eta_s = Im\left[\frac{1}{15}\rho\kappa l_p \left(\frac{-2i\varsigma_\perp}{\kappa}\right)^{3/4} \omega^{\frac{3}{4}}\right] \quad \text{(Eq. 3)}$$

where the area density of micelles $\rho = \frac{\phi}{\pi d^2/4}$,

the bending modulus $\kappa = k_B T l_p$,

and the lateral drag coefficient $\varsigma_\perp = \frac{4\pi\eta_s}{\ln(0.6\xi/d)}$

5) Calculate $\langle L \rangle$ (m) from the correlation

$$\frac{G'_{min}}{G''_{min}} = 0.317\left(\frac{\langle L \rangle}{l_e}\right)^{0.82} \quad \text{(Eq. 4)}$$

using l_e as calculated from $\alpha \equiv \frac{l_e}{l_p}$

6) Calculate $\xi \equiv \tau_{br}/\tau_{rep}$ from the correlation

$$\tau_R = \frac{1}{\omega_{c1}} = 0.484\tau_{br}^{0.63}\tau_{rep}^{0.37} \quad \text{(Eq. 5)}$$

where the reptation time $\tau_{rep} = \frac{2\langle L\rangle^3}{\pi^2 \alpha D_0}$,

in which the translational diffusivity within the tube $D_0 = \frac{k_B T}{\varsigma}$

and the drag coefficient $\varsigma = \frac{2\pi\eta_s}{\ln(\xi/d)}$

[Note: We have provided an excel spreadsheet that can be used to aid in calculating the micelle parameters as outlined above.]

Running a pointer algorithm simulation

The pointer algorithm can be compiled using the gfortran compiler and run locally in a command line or IDE, or run remotely on a computing cluster.

Understanding the output files

RESULT.DAT – At the end of the simulation, this file will contain the final extracted micelle parameters and G' and G" curves. For a noniterative pointer algorithm simulation, the parameters should match the inputs and the G' and G" curves are the predicted rheology. For an iterative simulation, the output is either the converged results or the results from the final (unconverged) iteration, in which case you may decide to restart the simulation with the final micelle parameters.

result_fit.dat – After every iteration, the simulation checks the error between the G' and G" curves from the current iteration and the experimental data. If the error has decreased from previous iterations, the current best-fit parameters and rheology curves are recorded in this file.

GF_t.DAT – This file contains the stress relaxation curve from reptation and its fit using the genetic algorithm assuming multiexponential relaxation of the form $\mu(t) = \sum_i \mu_i e^{t/\tau_i}$. These data are normalized by the plateau modulus and are in the form [time (s), μ(t) from simulation, μ(t) fitted with the genetic algorithm].

GW.DAT – The contents of the above file transformed into the frequency domain, scaled by the plateau modulus. The columns of this file are [ω (rad/s), G'(ω) (Pa), G"(ω) (Pa)] with $G'(\omega) = \sum_i G_0 \mu_i \frac{\omega\tau_i}{1+\omega^2\tau_i^2}$ and $G''(\omega) = \sum_i G_0 \mu_i \frac{(\omega\tau_i)^2}{1+\omega^2\tau_i^2}$.

rotary.dat, rouse_u.dat, bending_u.dat, rouse.dat, bending.dat – These files contain contributions to relaxation from rotary relaxation, Rouse modes of unentangled micelles, bending modes of unentangled micelles, Rouse modes of entangled micelles, and bending modes of entangled micelles respectively. All of these files contain data in the format [ω (rad/s), G'(ω) (Pa), G"(ω) (Pa)].

NEW_INPUT.DAT – This file tracks the best-fitting iteration and creates a new input file with those parameters in case the simulation does not converge and you want to restart it from the best-fit parameters. Note that it does not contain the experimental data that were input.

result_in.dat – The predicted G' and G" values are output to this file at the same frequencies as the experimental data for the best-fit iteration. Certain features of the experimental and pointer algorithm rheology such as the first crossover frequency are also output to this file. This information can be useful to directly compare the pointer algorithm predictions to the experimental data at the same frequencies.

Other files:

SIMULATION OUTPUT.DAT – This is an intermediate file that records the micelle parameters and the calculated error between the pointer algorithm G' and G" curves and the experimental data at each iteration. If the simulation is not converging, this file may be useful. You can look for iterations with lower error and restart the simulation from those parameters as an alternative to using the simulation-determined best-fit parameters.

INTRADATA.DAT – This file contains the micelle parameters and G' and G" curves for every iteration. It can be used to manually compare any iteration to the experimental data after potential iterations of interest have been identified in the SIMULATION OUTPUT file.

TEMP.DAT – The unrelaxed fraction of micelles $\mu(t)$ is written to this file in the format [time step (#), time (s), $\mu(t)$].

TIME_FREQUENCY TRANSFORMATION.DAT – This file has the results from fitting $\mu(t)$ to a multiexponential expression with the genetic algorithm. The data pairs are [μ_i, τ_i].

SIMULATION MONITOR.DAT – This file tracks the step of the pointer algorithm currently being performed (reptation, the genetic algorithm, etc.).

Example simulations

Example 1: An iterative pointer algorithm simulation

This example shows how to set up an iterative pointer algorithm simulation and the simulation results. The solution considered is a SLE1S + CAPB/NaCl solution with merged mechanical and high-frequency DWS rheology shown in Figure 3 below. The rheology was measured at 25°C, the salt solution (solvent) has a viscosity of η_s = 0.0011 Pa.s, the zero shear viscosity of the surfactant solution was measured to be 26.1 Pa.s, and the surfactant concentration was calculated to give a volume fraction of 0.06.

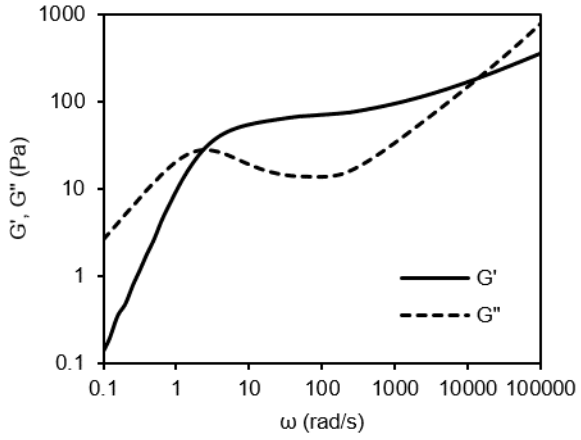

Figure 3: Experimental rheology for example SLE1S/CAPB solution.

Along with those known experimental parameters, we also need to estimate starting values for the micelle parameters and select options for the simulation. In this case, if we opt not to pre-calculate the micelle parameters before beginning the simulation, we only need to make guesses for the starting persistence length and the micelle length. For the persistence length, because high-frequency data are available, we choose to have the simulation fit the persistence length and estimate a starting value of 80 nm, close to values we previously found for other SLE1S/CAPB solutions. For the micelle length, the experimental data do show a minimum in G", but the ratio of G' to G" at the frequency where G" has its minimum is only moderately high (~5) so we don't expect the micelle length to be extremely long and guess 3 μm as a starting value for the micelle length. After choosing starting values for the micelle and persistence lengths, we can generate the input file for the simulation, shown in Figure 4.

In Figure 4 below, the experimental parameters T, ϕ, η_s, and η_0 (units given in the "preparing the simulation input file" section) are entered in lines 1 and 7 (column 1). The estimated micelle parameters, the persistence length and micelle length, are inputs 6.2 and 7.2 respectively. If we had also wanted to estimate the other independent fitting parameters α and ζ, those would have been entered in line 9 (and Y entered for input 9.1).

Line 3 contains the information about the form of the experimental data. Input 3.1 (Y) signifies that experimental rheological data are available, and 3.2 (N) means that high-frequency data are present. Since we entered Y for all items in line 4, the simulation will generate the output files GF_t.DAT, GW.DAT, and a file containing the contribution to relaxation from unentangled micelles added to the data in GW.DAT for the most recent iteration. See the above section for more information on the contents of GF_t.DAT and GW.DAT.

Input 5.1 is the switch for generating separate output files for the contributions to relaxation from Rouse, bending, and rotary modes for entangled and unentangled micelles. For the example simulation in Figure 4, these files will be output. Input 5.2 (Y) tells the simulation to iterate and fit the experimental data, not run a single-iteration predictive pointer algorithm simulation. The last input on line 5 (N) means that the simulation will not try to fit the stress relaxation curve from reptation with a single Maxwell mode, but will use the Genetic Algorithm to fit the stress relaxation curve with an appropriate number of modes.

As mentioned above, input 6.2 is the starting value of the persistence length in nanometers. Input 6.1 (Y) is the option for the pointer algorithm to fit the high-frequency data and extract the persistence length as a fitting parameter. We can choose this option because high-frequency data are present in this example.

The final input in line 7 is the micelle diameter in nanometers. We usually use a value of 4.0 nm here, but if you have a separate measurement of the micelle diameter is available from SANS or another experimental method, the diameter can be input here.

Line 8 contains the inputs for the number of micelles and number of iterations. From a series of simulations fitting experimental rheology with different ensemble sizes, we determined that 2000 micelles is the smallest ensemble size that gives the same results as larger ensembles. For the number of iterations, we find that 20 is a good number to allow enough iterations and time for the parameters to converge.

Line 9 was described above, and finally, lines 10 to the end of the file contain the experimental rheology in the order [ω(rad/s), G' (Pa), G"(Pa)].

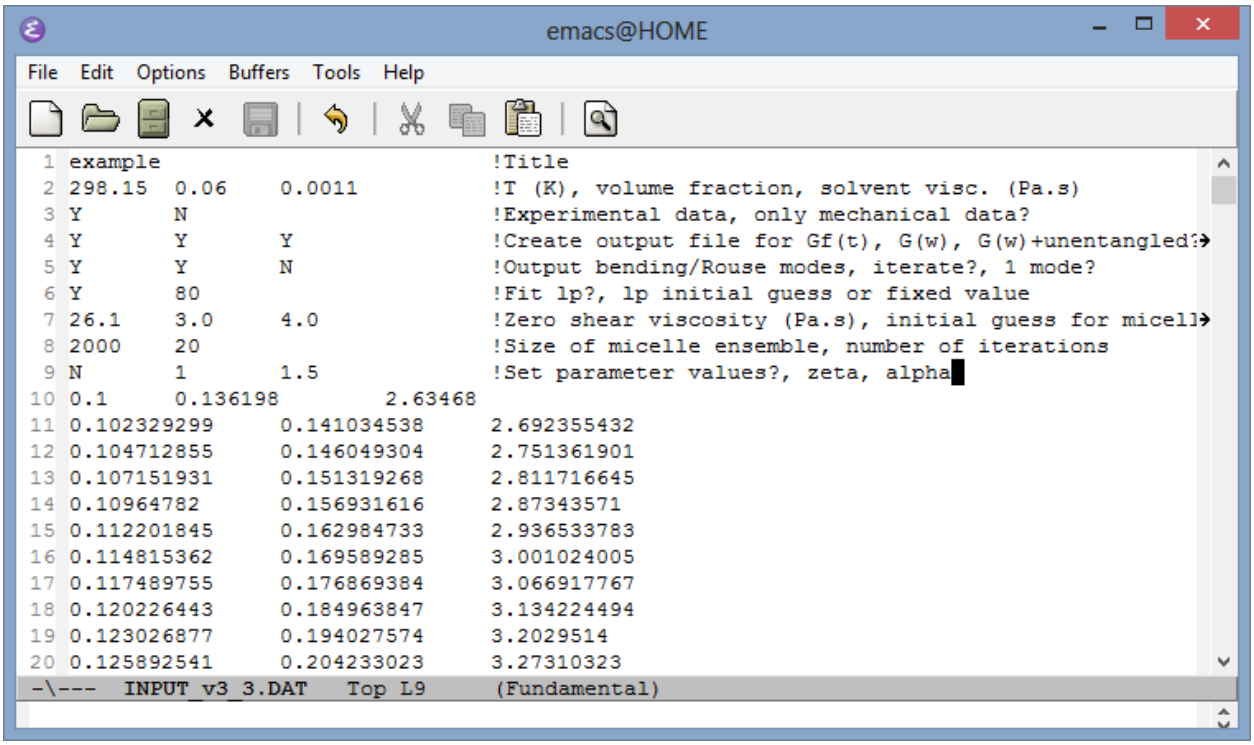

Figure 4: Example input file for data shown in Figure 3.

After completing the input file, we transferred the input file, the simulation code, and a submission script to the University of Michigan's high-performance computing cluster. [The simulations can be run locally, but their length often makes it more practical to run them remotely.] The simulation was given 5 days of wall time, but converged after 16 hours and the results are shown in Figure 5 and Table 1.

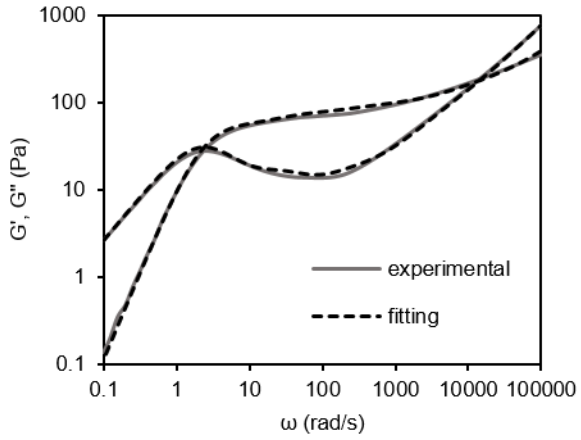

Figure 5: Example simulation results showing the fitted pointer algorithm rheology curves compared to the experimental data.

Table 1: Fitted micelle parameters extracted from experimental rheology

parameter	value
ζ	0.2
τ_{rep} (s)	3
τ_{br} (s)	0.4
$\langle L \rangle$ (μm)	3.7
G_0 (Pa)	98
l_e (nm)	140
l_p (nm)	91
$\alpha \equiv l_e/l_p$	1.6
$Z \equiv \langle L \rangle / l_e$	26

From Figure 5, we see good agreement between the pointer algorithm results and the experimental data.

Example 2: A predictive pointer algorithm simulation

If in the above example we had chosen to pre-calculate our micelle parameters, thought the micelles were very long, or didn't want to wait for an iterative simulation to converge, we could instead have run a predictive pointer algorithm simulation. To run a predictive simulation, we need to have better estimates of the micelle parameters, which can be obtained from the correlations in the "Calculating micelle parameters" section.

First, we need the experimental parameters T = 298 K, ϕ = 0.06, and η_s = 0.0011 Pa.s and the experimental rheological parameters G'_{min}, G''_{min}, and ω_{c1}. For this data set, G'_{min} = 69.9 Pa, G''_{min} = 13.8 Pa, and ω_{c1} = 2.34 rad/s.

The ratio G'_{min}/G''_{min} = 5.06 is less than 10 so Eq. 1 is used to calculate G_0 and

$$G_0 = G'_{min}\left(\frac{4.25}{G'_{min}/G''_{min}} + 0.625\right) = 69.9\ Pa\left(\frac{4.25}{69.9\ Pa/13.8\ Pa} + 0.625\right) = 102\ Pa$$

Next, because we want to extract the persistence length from the high-frequency rheology, Eqs. 2 and 3 must be solved simultaneously. Here we use an Excel spreadsheet that has been set up for this purpose. Part of the spreadsheet is shown in Figure 6. Columns M, N, and O are the experimental data. Column P is the right-hand side (RHS) of Eq. 3, calculated from the experimental data and column R is the LHS of Eq. 3, initially calculated from the known micelle parameters and placeholder values for l_e and l_p. We're interested in the high-frequency data when the slope is 0.75, so we calculate the slope of column P on a log-log plot in column Q. We

look for where the slope of the RHS of Eq. 3 is close to 0.75 and sum the differences (error) between columns P and R, the left and right sides of Eq. 3. The spreadsheet is also simultaneously calculating the plateau modulus from Eq. 2 and comparing it to G_0 determined earlier from Eq. 1, 102 Pa. We then use solver to minimize the two sources of error (fitting to high frequency data and the already calculated G_0) by varying l_e and l_p. The solver solution for this experimental data set is l_e = 141 nm and l_p = 92.5 nm, so $\alpha = \frac{l_e}{l_p} = \frac{141\ nm}{92.5\ nm} = 1.52$.

M	N	O	P	Q	R	S
Experimental data			RHS			
ω (rad/s)	G' (Pa)	G" (Pa)	G"-ηs.ω (Pa)	slope	LHS	error
75857.76	322.3717	631.219	545.1963215	0.727341	545.6441	-0.4478
77624.71	324.9301	642.4478	554.4214016	0.728707	555.1489	-0.7275
79432.82	327.5093	653.8931	563.8163013	0.729764	564.8192	-1.00294
81283.05	330.1099	665.5613	573.3863309	0.730971	574.658	-1.2717
83176.38	332.733	677.4594	583.1374097	0.732357	584.6682	-1.5308
85113.8	335.3766	689.5866	593.0675951	0.733329	594.8528	-1.78517
87096.36	338.0445	701.9581	603.1908119	0.735052	605.2147	-2.02391
89125.09	340.7325	714.5666	613.4987032	0.735893	615.7572	-2.25847
91201.08	343.4428	727.4227	624.000669	0.737141	626.4833	-2.48261
93325.43	346.1748	740.5302	634.6991273	0.738285	637.3962	-2.69709
95499.26	348.9317	753.9033	645.6071364	0.740042	648.4993	-2.89212
97723.72	351.7086	767.5323	656.7136413	0.740772	659.7957	-3.08206
100000	354.51	781.4361	668.036062	0.742387	671.2889	-3.25286
						0.009121

Figure 6: Portion of Excel spreadsheet showing fitting to Eq. 3, scrolled to the high-frequency data where the slope is close to 0.75.

$\langle L \rangle$ can now be calculated from Eq. 4, $\frac{G'_{min}}{G''_{min}} = 0.317\left(\frac{\langle L \rangle}{l_e}\right)^{0.82}$ and

$$\langle L \rangle = l_e\left[\frac{1}{0.317}\left(\frac{G'_{min}}{G''_{min}}\right)\right]^{\frac{1}{0.82}} = 0.141\ \mu m\left[\frac{1}{0.317}(5.06)\right]^{\frac{1}{0.82}} = 4.13\ \mu m$$

Lastly, the ratio of breakage to reptation time is determined from Eq. 5,

$\tau_R = \frac{1}{\omega_{c1}} = 0.484\tau_{br}^{0.63}\tau_{rep}^{0.37}$,

where the reptation time $\tau_{rep} = \frac{2\langle L \rangle^3}{\pi^2 \alpha D_0}$,

in which the translational diffusivity within the tube $D_0 = \frac{k_B T}{\varsigma}$

and the drag coefficient $\varsigma = \frac{2\pi\eta_s}{\ln(\xi/d)}$.

Starting from the drag coefficient and working upward, $\varsigma = \frac{2\pi(0.0011\ Pa.s)}{\ln(141^{0.6} * 92.5^{0.4}/4)} = 0.0021\ Pa.s$,

$$D_0 = \frac{1.38 * 10^{-23}\ J/K\ (298\ K)}{0.0021\ Pa.s} = 1.96 * 10^{-18}\frac{m^3}{s}$$

$$\tau_{rep} = \frac{2(4.13 * 10^{-6}\ m)^3}{\pi^2(1.52)(1.96 * 10^{-18}\ m^3/s)} = 4.80\ s$$

$$\tau_{br} = \left(\frac{1}{2.34/s\ (0.484)4.80\ s^{0.37}}\right)^{1/0.63} = 0.326\ s$$

and $\zeta = \frac{\tau_{br}}{\tau_{rep}} = \frac{0.326\ s}{4.80\ s} = 0.0678.$

With the independent micelle parameters calculated, the input file can be filled in much like the example above except with the calculated values in inputs 6.2, 7.2, 9.2, and 9.3 (persistence length, micelle length, zeta, and alpha respectively). Additionally, make sure Y is entered for input 9.1 so that the simulation uses the calculated values of zeta and alpha.

An example result is shown in Figure 7. Note that the calculated parameters are close to the fitted parameters given in Table 1, and although the simulation rheology curves do not match the experimental data as well as the fitted curves (Figure 5), the agreement is still reasonably good.

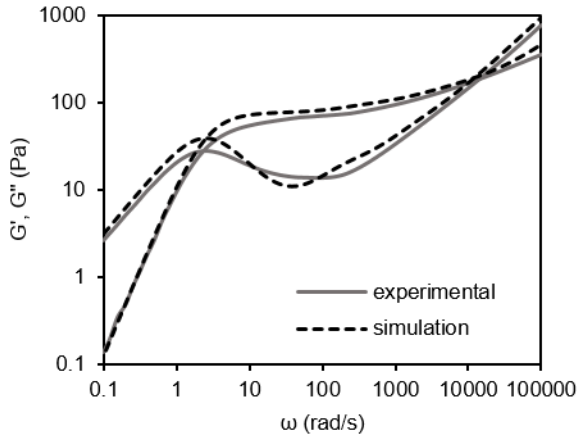

Figure 7: Rheology curves predicted by the pointer algorithm using micelle parameters calculated from correlations (Eqs. 1-5).

Troubleshooting

The simulation does not run:
If running a predictive pointer algorithm simulation (or a single iteration to compare to experimental data), the format or units of the input parameters may have been entered incorrectly, causing an error.

If running an iterative pointer algorithm simulation, you should still double check the input parameters, but there are also minimum requirements for the experimental data. The data need to contain a minimum in G" or if there is no minimum in G", high-frequency data that include the second crossover frequency must be available. Without either of these rheological features, an iterative fitting simulation cannot be run, but the micelle parameters can still be estimated from the rheology, as detailed in "Calculating micelle parameters."

The simulation does not finish:
For a predictive pointer algorithm simulation, you may just need to increase the wall time if running the simulation on a computation cluster. However, if the micelles are very long (greater than ~40-50 μm), even with an increased simulation time the simulation may take an excessive amount of time to finish. In this case, the number of micelles can be decreased to a minimum of ~500 micelles.

An iterative simulation may not finish because it also wasn't given enough time to complete all the iterations. Again, the wall time can be increased, but long micelles may make more than a few iterations difficult to complete in a reasonable amount of time. The options here are to increase the simulation time, use less iterations, or estimate the micelle parameters from the rheology and switch to a predictive simulation.

The simulation does not converge:
Although we have made an effort to test the pointer algorithm with a variety of surfactant/salt systems, the pointer algorithm may not be able to fit every single set of data. There may be a feature in your experimental data that the pointer algorithm cannot handle well. If the best-fit rheological curves look close to the experimental data, you can restart the simulation with the best-fit parameters and see if more iterations would allow the simulation to converge. If the best-fit G' and G" curves do not match the experimental data well, you may want to switch to a predictive simulation with the parameters calculated from the experimental data.

References

(1) Cates, M. E. Reptation of Living Polymers: Dynamics of Entangled Polymers in the Presence of Reversible Chain-Scission Reactions. *Macromolecules* **1987**, *20* (9), 2289–2296. https://doi.org/10.1021/ma00175a038.

(2) Wang, Z.; Larson, R. G. Molecular Dynamics Simulations of Threadlike Cetyltrimethylammonium Chloride Micelles: Effects of Sodium Chloride and Sodium Salicylate Salts. *J. Phys. Chem. B* **2009**, *113* (42), 13697–13710. https://doi.org/10.1021/jp901576e.

(3) Gittes, F.; Mackintosh, F. C. Dynamic Shear Modulus of a Semiflexible Polymer Network *Phys. Rev. E.* **1998**, *58* (2), 1241–1244. https://doi.org/10.1103/PhysRevE.58.R1241.

(4) Zou, W.; Larson, R. G. A Mesoscopic Simulation Method for Predicting the Rheology of Semi-Dilute Wormlike Micellar Solutions. *J. Rheol.* **2014**, *58* (3), 681–721. https://doi.org/10.1122/1.4868875.

(5) Zou, W.; Tang, X.; Weaver, M.; Koenig, P.; Larson, R. G. Determination of Characteristic Lengths and Times for Wormlike Micelle Solutions from Rheology Using a Mesoscopic Simulation Method. *J. Rheol.* **2015**, *59* (4), 903–934. https://doi.org/10.1122/1.4919403.

(6) Tang, X.; Zou, W.; Koenig, P. H.; McConaughy, S. D.; Weaver, M. R.; Eike, D. M.; Schmidt, M. J.; Larson, R. G. Multiscale Modeling of the Effects of Salt and Perfume Raw Materials on the Rheological Properties of Commercial Threadlike Micellar Solutions. *J. Phys. Chem. B* **2017**, *121* (11), 2468–2485. https://doi.org/10.1021/acs.jpcb.7b00257.

(7) Zou, W.; Tan, G.; Jiang, H.; Vogtt, K.; Weaver, M.; Koenig, P.; Beaucage, G.; Larson, R. G. From Well-Entangled to Partially-Entangled Wormlike Micelles. *Soft Matter* **2019**, *15* (4), 642–655. https://doi.org/10.1039/C8SM02223B.

(8) Sato, T.; Moghadam, S.; Tan, G.; Larson, R. G. A Slip-Spring Simulation Model for Predicting Linear and Nonlinear Rheology of Entangled Wormlike Micellar Solutions. *J. Rheol.* **2020**, *64* (5), 1045–1061. https://doi.org/10.1122/8.0000062.

(9) Tan, G.; Zou, W.; Weaver, M.; Larson, R. G. Determining Threadlike Micelle Lengths from Rheometry. *J. Rheol. (N. Y. N. Y).* **2021**, *65* (1), 59–71. https://doi.org/10.1122/8.0000152.