Self-assembly of coarse-grained ionic surfactants accelerated by graphics processing units: Supplementary Information

David N. LeBard,[a‡] Benjamin G. Levine,[a‡], Philipp Mertmann,[b] Stephen A. Barr,[b] Arben Jusufi,[a] Samantha Sanders,[b] Michael L. Klein[*a], and Athanassios Z. Panagiotopoulos[*b]

Forcefield Parameters

Below are the force-field parameters for all surfactants studied in this work.

Table 1: Bonding Potential Parameters

Bond	K_{bond} (kcal / Å^2)	r_0 (Å)
CM-SU	22.00	3.63
CM-CM	12.32	3.64
CM-CT	12.32	3.65

Table 2: Angle Potential Parameters

Angle	K_{ang} (kcal / rad^2)	θ_0 (rad)
CM-CM-SU	2.200	3.107
CM-CM-CT	2.380	3.054
CM-CM-CM	2.380	3.019

[a] *Institute for Computational Molecular Science and Department of Chemistry, Temple University, 1900 North 12th Street, Philadelphia, PA, 19122, USA. Fax: +1 215 204 1532; Tel: +1 215 204 4212; E-mail: mlklein@temple.edu*

[b] *Department of Chemical and Biological Engineering and Institute for the Science and Technology of Materials, Princeton University, Princeton, NJ, 08544, USA; E-mail: azp@princeton.edu.*

‡These authors contributed equally to this work

Table 3: Non-bonding Potential Parameters

Atom Types	Potential Form	ε (kcal)	σ (Å)	q
SU SU	U_{LJ}^{9-6}	0.700	4.3210	-0.1118
SU CM	U_{LJ}^{9-6}	0.383	4.4135	0.0000
SU CT	U_{LJ}^{9-6}	0.405	4.4530	0.0000
SU SOD	U_{LJ}^{12-4}	1.100	4.1000	0.0000
SU W	U_{LJ}^{12-4}	1.100	4.1000	0.0000
CM CM	U_{LJ}^{9-6}	0.420	4.5060	0.0000
CM CT	U_{LJ}^{9-6}	0.444	4.5455	0.0000
CM SOD	U_{LJ}^{12-4}	0.340	4.4385	0.0000
CM W	U_{LJ}^{12-4}	0.340	4.4385	0.0000
CT CT	U_{LJ}^{9-6}	0.469	4.5850	0.0000
CT SOD	U_{LJ}^{12-4}	0.360	4.4780	0.0000
CT W	U_{LJ}^{12-4}	0.360	4.4780	0.0000
SOD SOD	U_{LJ}^{12-4}	0.350	4.3710	0.1118
SOD W	U_{LJ}^{12-4}	0.895	4.3710	0.0000
W W	U_{LJ}^{12-4}	0.895	4.3710	0.0000

HOOMD-Blue example input file

Below is an example input file for SDS (1 M) that runs according to the HOOMD-Blue 0.9.2 specification for ∼24ns with a 9.8 fs timestep. The parameters for all force constants, minimum bond/angle lengths, well-depth energies, charges, and particle sizes are provided in the tables above.

```
#! /usr/bin/env hoomd
# SDS @ 1M
num="0001"
prefix="sds."
dcdfilename=prefix+num+".dcd"
logfilename=prefix+num+".log"
xmlfilename=prefix+num+".xml"

# import hoomd environment
from hoomd_script import *
import math;
import os;

# setup the simulation
init.read_xml(filename='sds.0000.xml')

# xml file not provided, but can be created from a
# configuration file of a different format (e.g. pdb)
# using VMD with the HOOMD molfile plugin, available at:
# http://sites.google.com/site/akohlmey/software/hoomd-plugin
# and topotools v1.1 or greater, available at:
# http://sites.google.com/site/akohlmey/software/topotools

# note that HOOMD does calculations in a unitless fashion.
# We work in the following consistent set of reduced units:
# sigma = 1 Angstrom = 1e-10 m
# epsilon = 1 kcal / mol = 6.947701e-21 J = 6.947701e-21 kg m^2 / s^2
# mass = 1 AMU = 1.66056e-27 kg
# time = 48.889 fs
# kt = 1.9872065e-3 (kcal/mol)/K
# temp = 503.219 K
# pressure = 1 kcal/mol / A^3 = 68568.48 atm
# surface tension = 6.947701e2 mN / m
# k_e = 1 / ( 4 pi epsilon_0 ) = 332.06382 kcal/mol * A / electron^2
# charge = sqrt( 1 kcal/mol * A / k_e ) = .05487686 electron

# force field setup (see table and Equations 1-6
# in the text for a detailed explaination
nb = pair.cgcmm(r_cut=15.0)
```

```
nb.pair_coeff.set('SU', 'SU', epsilon=0.7000, sigma=4.3210,
                  alpha=1.0, exponents='LJ9-6')
nb.pair_coeff.set('SU', 'CM', epsilon=0.3830, sigma=4.4135,
                  alpha=1.0, exponents='LJ9-6')
nb.pair_coeff.set('SU', 'CT', epsilon=0.4050, sigma=4.4530,
                  alpha=1.0, exponents='LJ9-6')
nb.pair_coeff.set('SU', 'SOD', epsilon=1.1000, sigma=4.1000,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('SU', 'W', epsilon=1.1000, sigma=4.1000,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('CM', 'CM', epsilon=0.4200, sigma=4.5060,
                  alpha=1.0, exponents='LJ9-6')
nb.pair_coeff.set('CM', 'CT', epsilon=0.4440, sigma=4.5455,
                  alpha=1.0, exponents='LJ9-6')
nb.pair_coeff.set('CM', 'SOD', epsilon=0.3400, sigma=4.4385,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('CM', 'W', epsilon=0.3400, sigma=4.4385,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('CT', 'CT', epsilon=0.4690, sigma=4.5850,
                  alpha=1.0, exponents='LJ9-6')
nb.pair_coeff.set('CT', 'SOD', epsilon=0.3600, sigma=4.4780,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('CT', 'W', epsilon=0.3600, sigma=4.4780,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('SOD', 'SOD', epsilon=0.3500, sigma=4.3710,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('SOD', 'W', epsilon=0.8950, sigma=4.3710,
                  alpha=1.0, exponents='LJ12-4')
nb.pair_coeff.set('W', 'W', epsilon=0.8950, sigma=4.3710,
                  alpha=1.0, exponents='LJ12-4')

bnd = bond.harmonic()
bnd.set_coeff('CM-SU', k=22.0000, r0=3.6300)
bnd.set_coeff('CM-CM', k=12.3200, r0=3.6400)
bnd.set_coeff('CM-CT', k=12.3200, r0=3.6500)

ang = angle.cgcmm()
ang.set_coeff('CM-CM-SU', k=2.200, t0=3.107,
              epsilon=0.3830, sigma=4.4135, exponents='LJ9-6')
ang.set_coeff('CM-CM-CT', k=2.380, t0=3.054,
              epsilon=0.4440, sigma=4.5455, exponents='LJ9-6')
ang.set_coeff('CM-CM-CM', k=2.380, t0=3.019,
              epsilon=0.4200, sigma=4.5060, exponents='LJ9-6')
nlist.reset_exclusions(exclusions = ['bond', 'angle'])

pppm=charge.pppm(group=group.all())
```

```
pppm.set_params(Nx = 64, Ny = 64, Nz = 64, order = 4, rcut = 13.15)

group_surf = group.tags(name="surfactants", tag_min=0, tag_max=2999)

sorter.set_params(grid=32)

logdata  = ['time', 'potential_energy', 'temperature',
            'kinetic_energy', 'volume']

outchunk = 20000

log = analyze.log(filename=logfilename,
      quantities=logdata, period=outchunk, header_prefix='#')

# dump out only surfactant/counterion data to the dcd file
dcd = dump.dcd(filename=dcdfilename, period=outchunk,
               group=group_surf, overwrite=True)
xml = dump.xml()
xml.set_params(all=True)
all = group.all()

# reduced units are constructed with energy = 1 kcal/mol,
# sigma = 1 Angstrom, and mass = 1 amu as basic units
#
# T* = 0.602124 = T (303 K)
# P* = 0.00001458396 = P (1 atm)
integrate.npt(group=all,T=0.602124,tau=2000.0,
              P=0.00001458396,tauP=2000.0)

# t* = 0.20 = t (9.8 fs)
integrate.mode_standard(dt=0.20)
run(2400000)

xml.write(filename=xmlfilename)
```