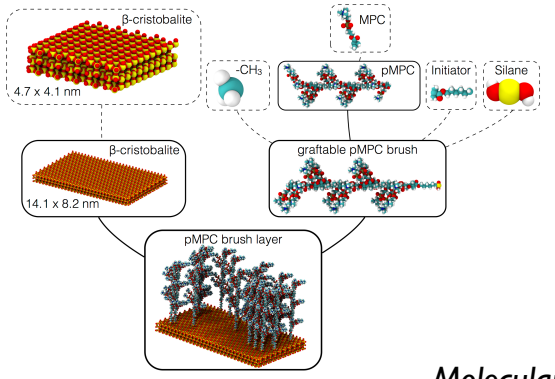
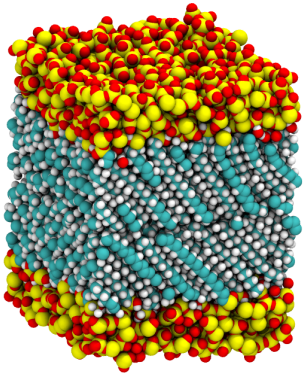
Computational Screening of Soft Materials Systems with Application to Nano-Lubrication Systems

Peter T. Cummings

Department of Chemical and Biomolecular Engineering, Vanderbilt University, Nashville, TN

Vanderbilt University Multiscale Modeling and Simulation (MuMS) Center, Nashville, TN





SCHOOL OF ENGINEERING

Telluride Science Research Center Workshop

Molecular engineering of soft matter: Spanning small molecules to macromolecules

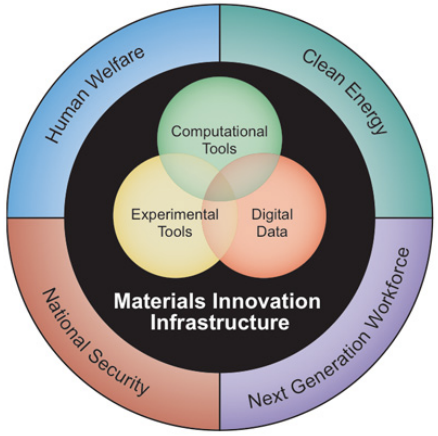
Telluride, CO

June 19-24, 2017

MUMS

Motivation - Computational Materials Discovery

- Computational discovery of new materials is emerging as new subfield of computational materials science developing out of the Materials Genome Initiative (MGI), e.g.:
  - **The Materials Project**
    - https://materialsproject.org
    - Properties of over 80,000 materials and screened 25,000 of these for Li-ion batteries.
      - *Based on use of low-level electronic structure methods on crystalline (ordered) materials*
  - **Screening of metal-organic frameworks (MOFs)[1]**
    - 137,953 hypothetical MOF structures generated
      - *Each MOF evaluated for methane storage at 35 bar and 298 K using grand canonical Monte Carlo*
    - ~300 MOFs had higher methane-storage capacity at 35 bar than current world record
  - **Crowdsourced Clean Energy Project database at H**
    - http://www.molecularspace.org
    - Data and analyses on 2.3 million candidate compounds for organic photovoltaics
      - *Ab initio methods applied to single molecules, linking structural and electronic properties*



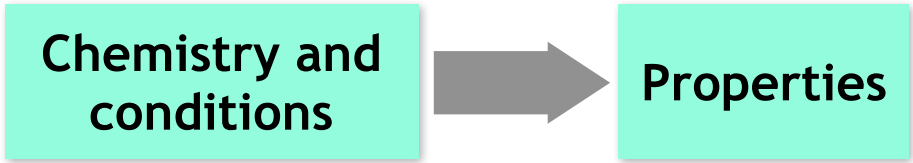
https://www.mgi.gov/sites/default/files/documents/materials_genome_initiative-final.pdf

Chemistry and conditions → **Properties**

[1] Wilmer, et al., Large-scale screening of hypothetical metal–organic frameworks. Nature Chemistry, 4 (2011) 83-89. doi:10.1038/nchem.1192

Motivation - MGI for Soft Matter

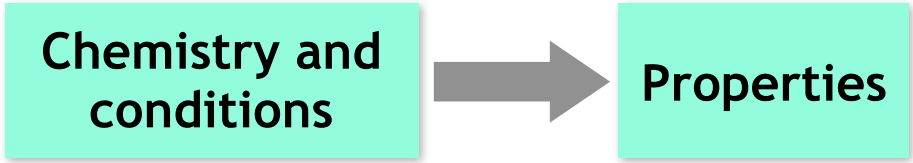
- In soft matter, predominant physical behaviors occur at energy scales comparable with k_BT
  Quantum aspects generally unimportant
  - Weak dispersion interactions frequently dominate
    - Contrast with hard materials
      - *Energy scales large by comparison to k_BT*
      - *Relatively easy to predict properties since atoms/molecules are typically in crystalline lattice*
  - It can be difficult to predict properties directly from atomic or molecular constituents
    - Soft matter often self-organizes into mesoscopic physical structures that are much larger than microscopic scale yet much smaller than the macroscopic scale
      - *Macroscopic properties are result of mesoscopic structure*
      - *Hard materials typically have no mesoscopic structuring*
    - Many steps are needed to gather properties based on a given chemistry and condition



MoSDeF

The Molecular Simulation and Design Framework

- Motivation - develop a platform for performing MGI-style screening of soft-matter systems
  - Goal:

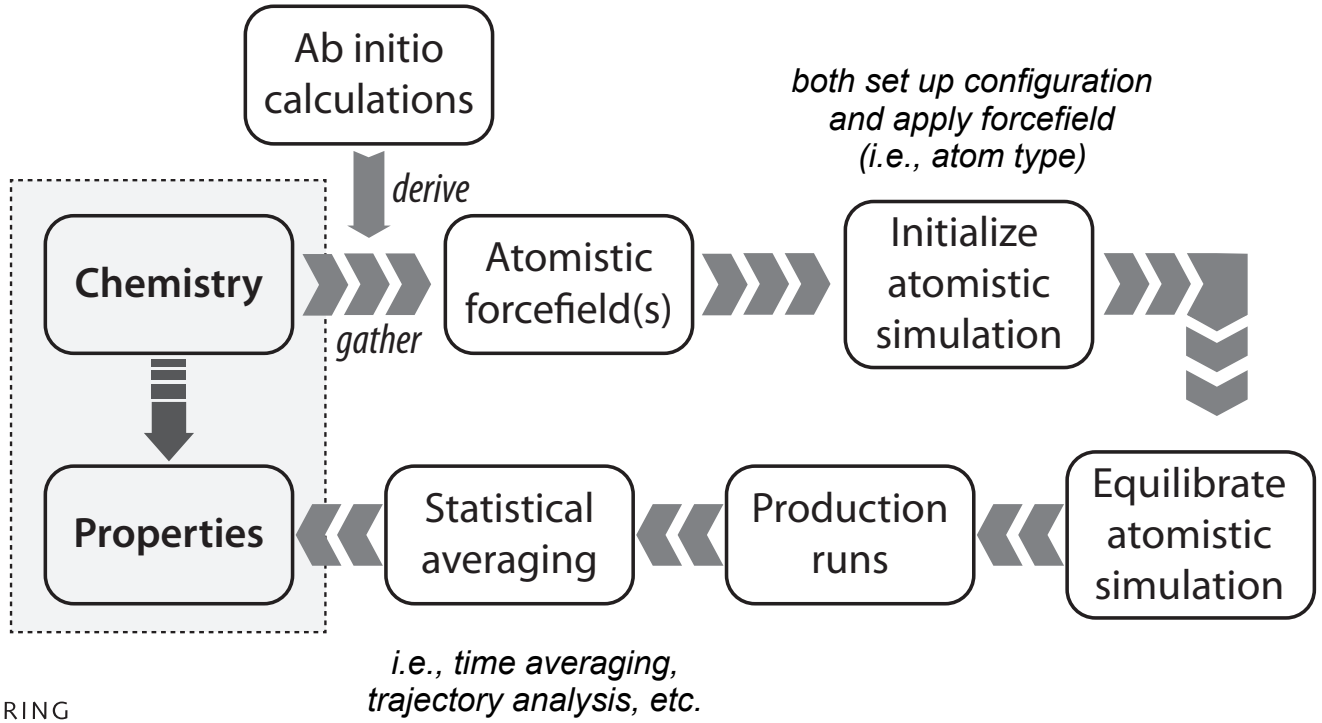


  - The procedures grouped into the arrow can be considered a "cottage industry" that is not always easy to reproduce
    - Only a few people have the knowledge to reproduce the specific simulation
      - *Procedures are not always well documented/described*
      - *The properties may depend on the exact details of the procedures*
    - *Ad hoc* tools/scripts are not always made available
      - *Software may be highly specific and not extensible*
      - *Software may be poorly written and not be well validated*

MoSDeF

A Molecular Simulation and Design Framework

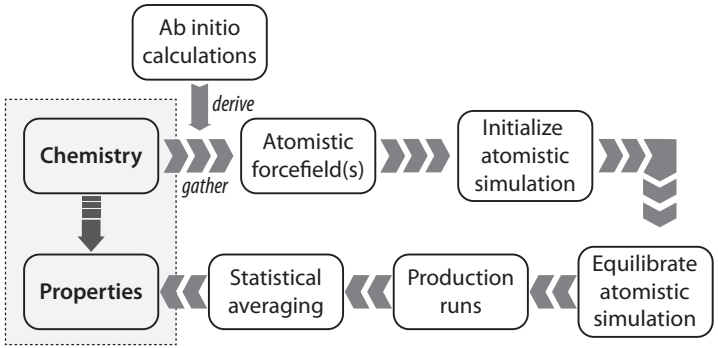
- **Motivation** - develop a platform for performing Materials Genome Initiative (MGI) style screening of soft-matter systems



MoSDeF

A Molecular Simulation and Design Framework

- **Motivation** - develop a platform for performing Materials Genome Initiative (MGI) style screening of soft-matter systems
  - To do so, need to automate each of these steps and be able to combine them together
    - Model-Integrated Computing (MIC)
    - Developed by Vanderbilt researchers in the Institute for Software Integrated Systems (ISIS)
    - Used in wide variety of applications where scientific/engineering workflows require automation and integration to achieve complex goals
      - *Car manufacturing assembly line automation and control (GM)*
      - *Health records management (NIH)*
      - *Amphibious infantry fighting vehicle design (DARPA)*
    - MIC involves meta-language abstraction
  - MoSDeF is outgrowth of three NSF grants joint with ISIS
    - NSF CDI CBET 1028374, NSF SSI ACI 1047828, NSF SSE ACI 1535150
    - MoSDeF also seeks to address **simulation reproducibility**



Sztipanovits, J. Karsai, G., "Model-integrated computing," *Computer*, **30**, 110-111 (1997)

MoSDeF

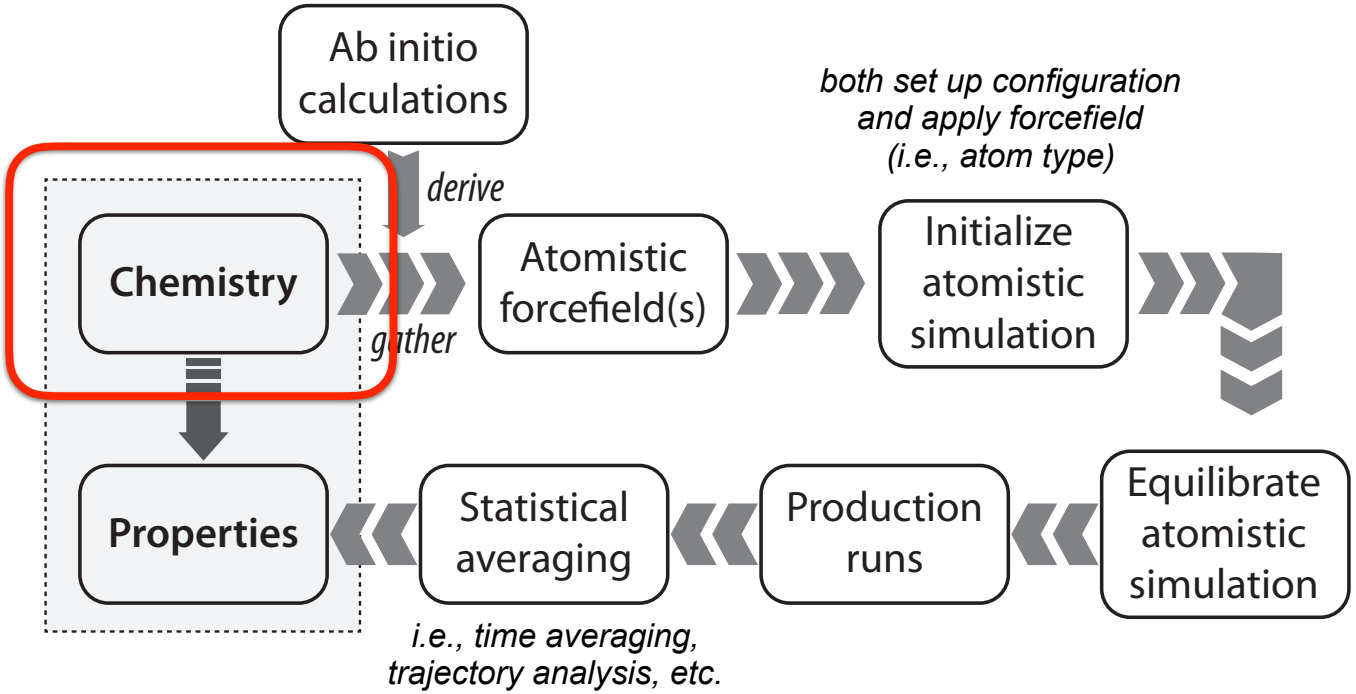
A Molecular Simulation and Design Framework

- Why does MoSDeF promote simulation reproducibility and why is this important?
  - In a science-skeptic world, the "reproducibility crisis" in scientific research has emerged as critical issue
    - http://www.nature.com/news/reproducibility-1.17552
  - Reproducibility in computational science is becoming a forefront issue in the computational science community
    - Automation is a key device for reproducibility
      - *Sandve, G. K. et al. (2013), Ten simple rules for reproducible computational research, PLOS Comp. Bio. (editorial), Vol. 9(10):1–4, doi: 10.13711/journal.pcbi.1003285*
    - "Reproducible computational research, in which all details of computations—code and data—are made conveniently available to others, is a necessary response to [the credibility] crisis."
      - *Donoho, D. et al. (2009), Reproducible research in computational harmonic analysis, Comp. Sci. Eng. 11(1):8–18, doi: 10.1109/MCSE.2009.15*
    - GUIs are the enemy of reproducibility
      - *Lorena A. Barba, George Washington University; see her blog http://lorenabarba.com/category/blog/ and "The hard road to reproducibility", Science, 354(6308):142 (October 2016)*
  - Scriptability, open-source, complete data dissemination needed to reproduce published simulations are goals inherent to MoSDeF

MoSDeF

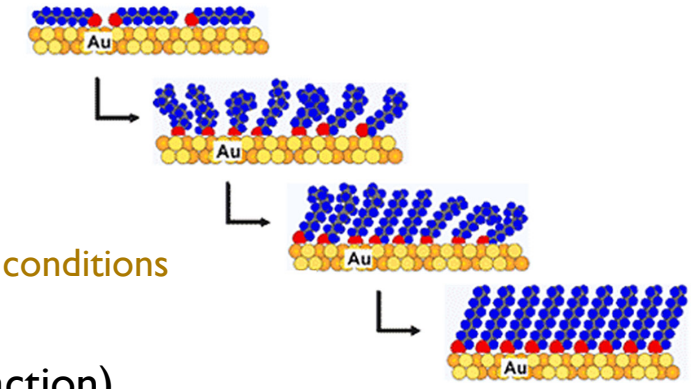
A Molecular Simulation and Design Framework

- **Motivation** - develop a platform for performing Materials Genome Initiative (MGI) style screening of soft-matter systems



Missing Functionality in Existing Tools

- Surface functionalization
  - Need ability to express patterns
  - Expose parameters that tune chemistry
- Flexible force field development and dissemination
  - Few force fields exist for all chemistries we care about
  - Not certain how accurate existing FFs will behave under high shear conditions
  - Collaborators and group members developing for specific systems
- Desire to use different simulation engines (meta-level abstraction)
  - LAMMPS, HOOMD, Gromacs all currently used in group
- MoSDeF is open source, built on open source components

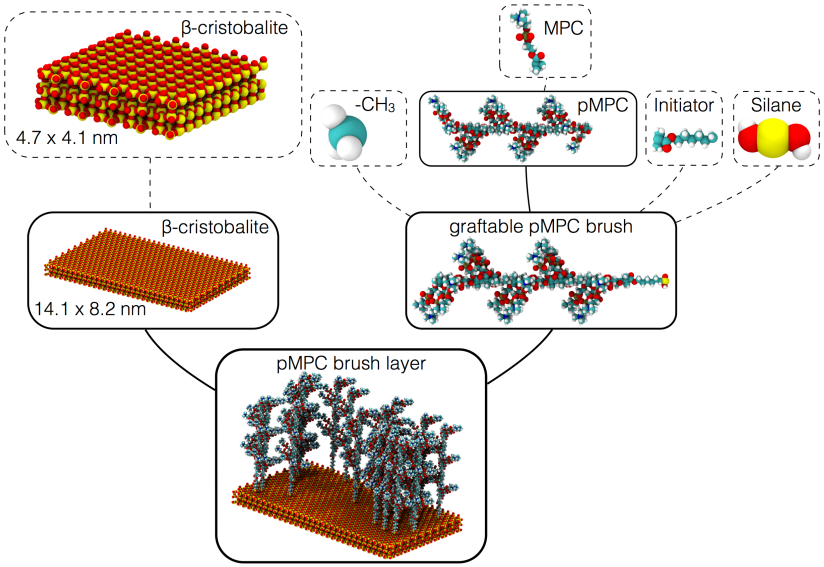


Nuzzo *et al.*, *JACS*, **1983**, *105*

NumPy SciPy matplotlib IPython Interactive Computing pandas scikit learn

mBuild: a Hierarchical Molecular Builder

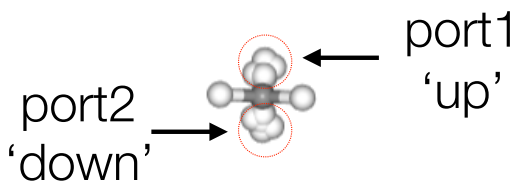
- Sketch or otherwise create simple components (dashed boxes)
- Combine operations on components into more complicated structures
  - Our applications (nanotribology, supercapacitors, skin lipid self-assembly) typically involve interfacial systems
- Everything is scriptable



https://github.com/mosdef-hub/mbuild.git

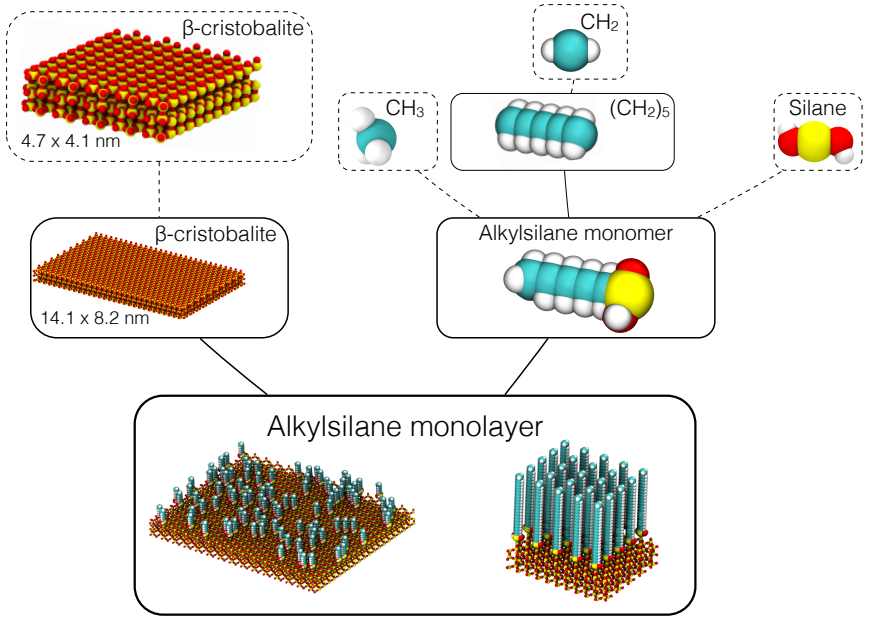
Example: polymerization

```
import mbuild as mb
from mbuild.lib.moieties import CH2

polymer = mb.Compound()

last_monomer = CH2()
polymer.add(last_monomer)

for _ in range(10):
    current_monomer = mb.clone(last_monomer)

    mb.force_overlap(move_this=current_monomer,
                     from_port=current_monomer['up'],
                     to_port=last_monomer['down'])

    polymer.add(current_monomer)
    last_monomer = current_monomer
```



- Above code with additional features is provided as mb.Polymer

Exposing Tunable Chemistry as Simple Variables




```
pattern = Random2DPattern(100)
chain_length = 5
n_tiles = [2, 3, 1]
```

```
pattern = Grid2DPattern(5, 5)
chain_length = 20
n_tiles = [1, 1, 1]
```

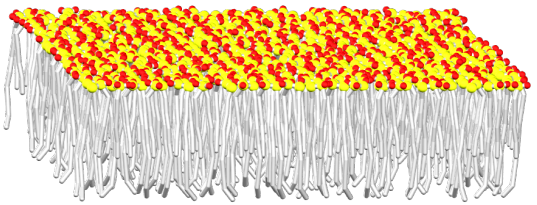
```
monolayer = AlkaneMonolayer(chain_length, pattern, n_tiles)
```

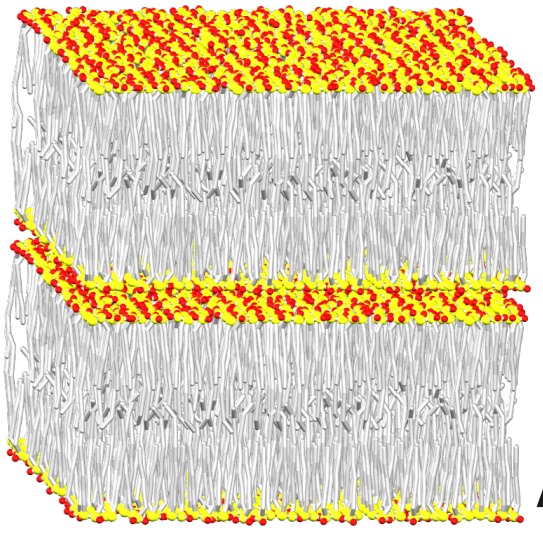

C. Klein, J. Sallai, T.J. Jones, C.R. Iacovella, C. McCabe, P.T. Cummings, A Hierarchical, Component Based Approach to Screening Properties of Soft Matter, in: Springer, Singapore, 2016: pp. 79–92. doi:10.1007/978-981-10-1128-3_5.

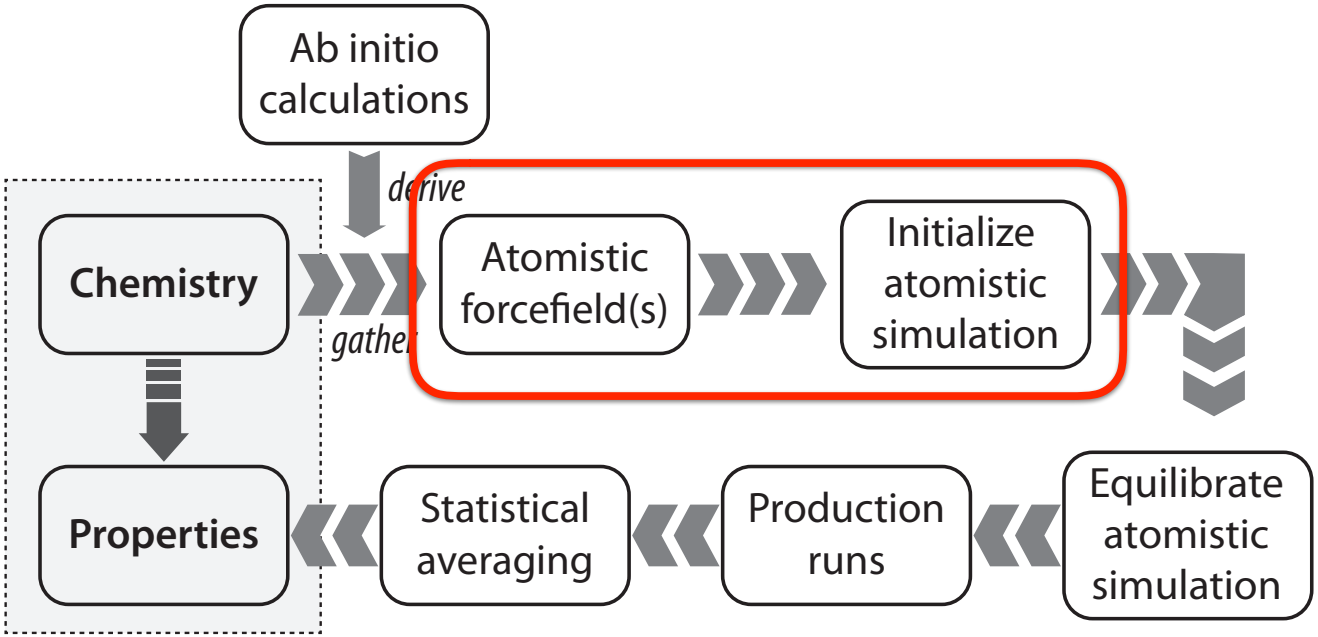
Using mBuild: CER Multilayer

- Easily build a ceramide multilayer with just a few lines of python
  - Read in molecule from pdb file
  - Build leaflet, then stack leaflets
  - 13 lines!

```
1  from numpy import pi
2  import mbuild as mb
3
4  cer2 = mb.load('cer2.pdb')
5  grid = mb.pattern.Grid2DPattern(50, 50)
6  leaflet, system = mb.Compound(), mb.Compound()
7  repeat_spacing = 6  # repeat spacing in nm
8  leaflet = grid.apply(cer2)  # put lipids on gird points
9  leaflet = mb.Compound(leaflet)
10 for layer_num in range(4):  # add 4 leaflets to the system
11     new_layer = mb.clone(leaflet)
12     if layer_num % 2 == 1:  # rotate every other layer 180deg
13         new_layer.spin(pi, around=[1, 0, 0])
14     new_layer.translate([0, 0, layer_num * repeat_spacing / 2])
15     system.add(new_layer)
16 system.save('cer2.hoomdxml')
```






Ab initio calculations
derive
Chemistry
Atomistic forcefield(s)
Initialize atomistic simulation
Properties
Statistical averaging
Production runs
Equilibrate atomistic simulation

Atom Typing and Applying Force Fields

- Need general purpose atom typer
  - Arbitrary chemistry means we cannot rely on templates
  - Needs to be easy to develop new atom types and associated logic
  - Don't want to rely on rigid rule hierarchies for atom type definitions
    - Ordered least to most general
- Removing gap between logic and parameters
  - Parameters exist in various files
  - Logic often encoded as series of if/else statements deep in source code
  - Why not combine the two?
- Similar concept being developed by Open Force Field group
  - https://github.com/open-forcefield-group/

Foyer: Applying and Disseminating force fields

- End user should only require
  - Force field file
  - Chemical topology

```
from foyer import Forcefield
import parmed as pmd

untyped_ethane = pmd.load_file('ethane.mol2', structure=True)
oplsaa = Forcefield(forcefield_files='oplsaa.xml')
ethane = oplsaa.apply(untyped_ethane)

# Save to any format supported by ParmEd
ethane.save('ethane.top')
ethane.save('ethane.gro')
```

Adding Atom Type Definitions to OpenMM XML

- OpenMM XML file convenient to extend
  - Existing infrastructure
  - Great support for custom functional forms

```
<ForceField>
 <AtomTypes>
  <Type name="opls_135" class="CT" element="C" mass="12.01100" def="[C;X4](C)(H)(H)H" desc="alkane CH3"/>
  <Type name="opls_140" class="HC" element="H" mass="1.00800"  def="H[C;X4]" desc="alkane H"/>
 </AtomTypes>
</ForceField>
```

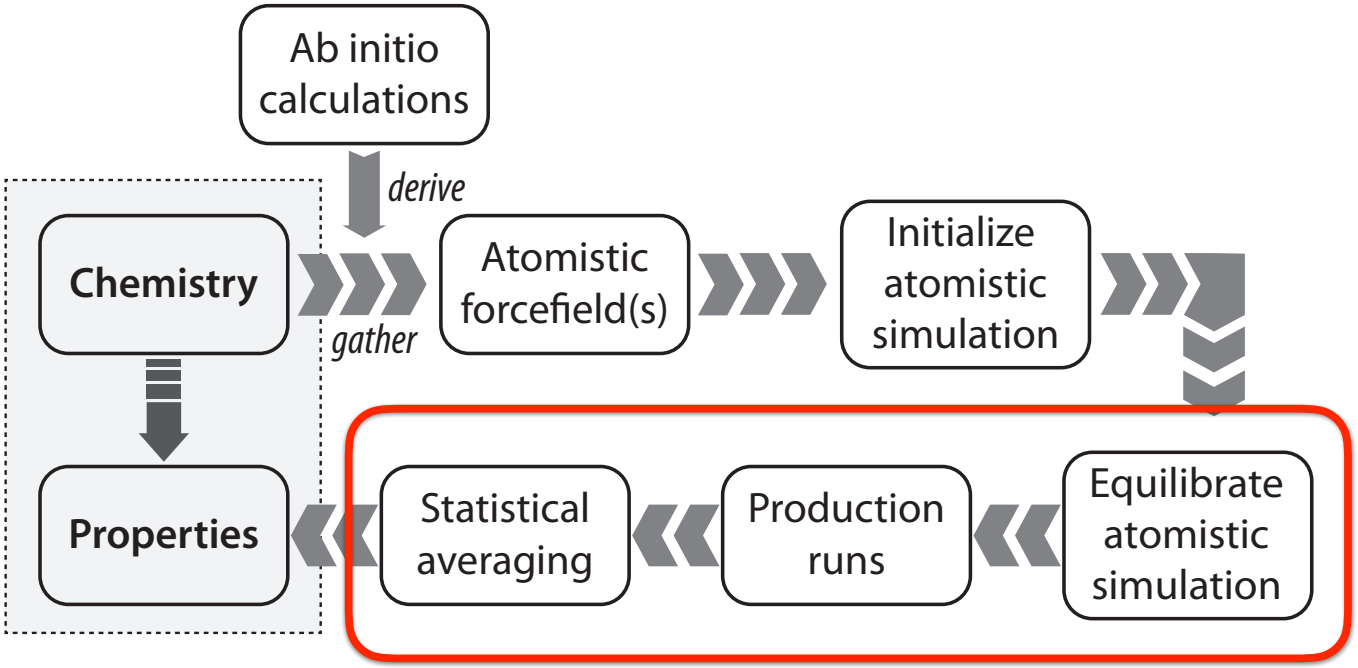
- Add machine/human readable SMARTS
  - First atom token in SMARTS indicates the type that we are defining
- Optionally, add colloquial description and DOI
  - Adding DOI produces BibTeX file when applying force field

Promoting Dissemination of Force Fields

- Template repo for building foyer compatible force field files
  - https://github.com/mosdef-hub/forcefield_template
  - Skeleton force field file
  - Skeleton testing setup
    - User needs to only add correctly typed .mol2 files
  - Tutorial for adding SMARTS definitions
    - Walkthrough for adding more and more complex definitions
    - Testing with small example set of hydro carbons from alkanes through alkenes and benzene
    - How to run tests with py.test and enable TravisCI for your repo
- Force field development within our group
  - **OPLS-AA compatible parameters for perfluoroethers**

    DOI 10.5281/zenodo.56807
  - Coarse-grained models for skin lipids
  - Automating force field derivation for supercapacitor electrodes

VANDERBILT


Ab initio calculations
derive
Chemistry
gather
Atomistic forcefield(s)
Initialize atomistic simulation
Properties
Statistical averaging
Production runs
Equilibrate atomistic simulation

MetaMDS - Overview

- MetaMDS was designed with the goal of being able to perform screening simulations
  - It can submit large batches of jobs with various input parameters
- MetaMDS relies on 3 key inputs:
  - mBuild "recipe"
    - This tells metaMDS how to construct the system
    - Allows us to screen over not just thermodynamic properties, but over different structures/molecules/etc.
  - Simulation template
    - This is a script that encapsulate the underlying simulation routines/calls
  - List(s) of parameters to vary
    - These parameters are used to change thermodynamic states in the simulation script and change inputs to the mBuild recipe
    - These lists could also include different forcefields to evaluate

MetaMDS - Screening

- By using simulation templates, mBuild recipes, and simple parameter lists, we can create a streamlined, abstracted interface
  - Users only need a few lines of code to do large scale, complex screening
  - Individual elements can be easily tested before large scale deployment
  - All procedures are well documented and described

In [ ]:
```
# Initialize a simulation instance with a template and some metadata
sim = mds.Simulation(name='monolayer', template=create_run_script, output_dir='output')

chain_lengths = [8, 12, 16, 20]
for length in chain_lengths:
    parameters = {'chain_length': length,
                  'n_molecules': 100,
                  'forcefield': 'OPLS-aa',
                  'build_func': build_monolayer}
    # Parameterize our simulation template
    sim.parametrize(**parameters)
```

In [ ]:
```
# Run
sim.execute_all(hostname='rahman.vuse.vanderbilt.edu', username='ctk3b')
```

In [ ]:
```
sim.sync_all()
```

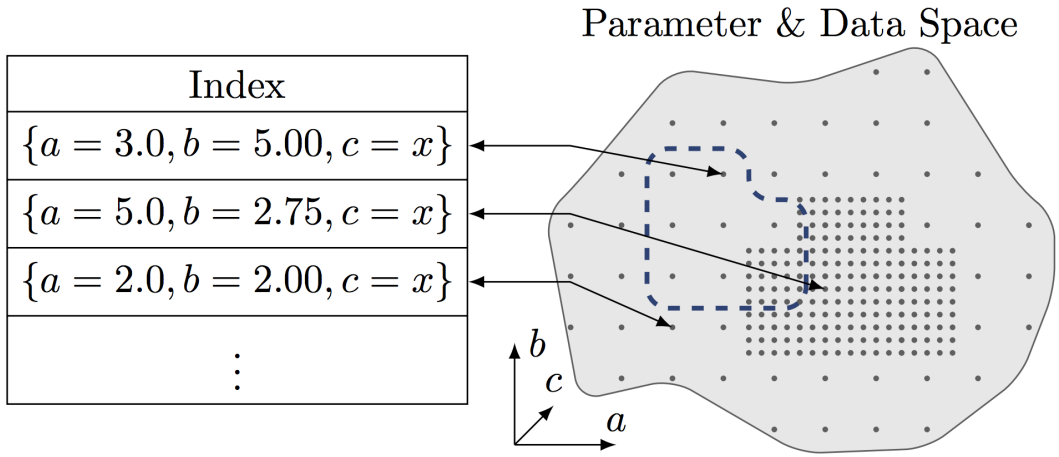

User interface is Python notebook (c.f. Mathematica)

MoSDeF

- Building a community
  - Hackathon in February with representation from 11 research groups and ExxonMobil
  - Goal of learning how to use MoSDeF and become co-developers
    - Add forcefields
      - *Currently support OPLS, TRAPPE, specific forcefields (e.g., ionic liquids)*
      - *Wish to add other standard forcefields (e.g., bio forcefields, ReaxFF generic and specialized) plus workflows to automate forcefield derivation*
    - Add simulation workflows for specific properties and applications
      - *Currently support many standard workflows, coarse-graining (MSIBI), etc*
      - *LONG wish list!*
    - Add simulation packages
      - *Currently support LAMMPS, GROMACS, HOOMD-Blue*
      - *Wish to add open-source AIMD codes, Monte Carlo (e.g., Cassandra), other MD codes*
  - Plan to jointly submit NSF SSI to support next MoSDeF cycle

Putting it All Together: Signac Workflow Manager

- Recently adopted in place of metaMDS
- Define heterogenous parameter spaces
- Provides link between data and metadata
  - Every job and its data is tied to a point in parameter space
  - Provides glue for seamless building, execution and analysis



Developed by Simon Adorf in Glotzer group

Adorf et al., arXiv:1611.03543 [cs.DB]

Putting It All Together

- Our tribology operations
  - mBuild + save with foyer
  - Minimize
  - Equilibrate
  - Shear at range of normal load
- Customize compute environment
  - Defaults provided for common envs
  - Collection of envs for major computing centers
    - OLCF, NERSC, NICS
- Execute and monitor

Putting It All Together

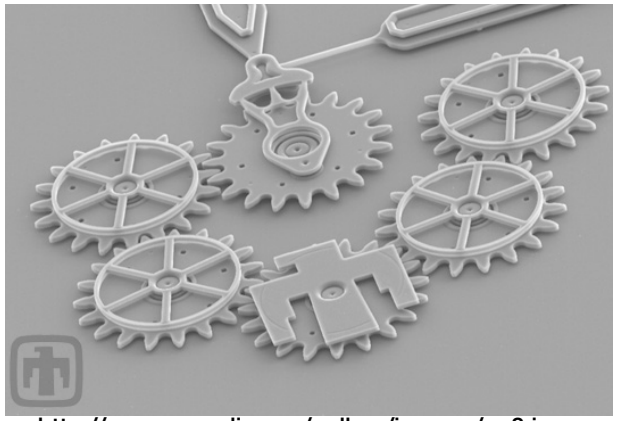
- Our tribology operations
  - mBuild + save with foyer
  - Minimize
  - Equilibrate
  - Shear at range of normal load
- Customize compute environment
  - Defaults provided for common envs
  - Collection of envs for major computing centers
    - OLCF, NERSC, NICS
- Execute and monitor

```
class MyTorqueEnvironment(flow.environment.TorqueEnvironment):
    hostname_pattern = 'mymoabcluster.university.edu'
    cores_per_node = 16

    @classmethod
    def mpi_cmd(cls, cmd, np):
        return 'mpirun -np {np} {cmd}'.format(n=np, cmd=cmd)

    @classmethod
    def script(cls, _id, nn, walltime, ppn=None, **kwargs):
        if ppn is None:
            ppn = cls.core_per_node
        js = super(MyTorqueEnvironment, cls).script()
        js.writeline('#!/bin/sh -l')
        js.writeline('#PBS -j oe')
        js.writeline('#PBS -l nodes={}:ppn={}'.format(nn, ppn))
        js.writeline('#PBS -l walltime={}'.format(format_timedelta(walltime)))
        js.writeline('#PBS -q low')
        js.writeline('#PBS -N {}'.format(_id))
        js.writeline('#PBS -V')
        return js
```

```
>>> project = MyProject()
>>> project.print_status(detailed=True, params=('a',))
Status project 'test-project':
Total # of jobs: 10
label         progress
------------  ------------------------------------------------
initialized   |######################################| 100.00%
processed     |########################--------------| 66.67%
Detailed view:
job_id                           S  next_job  a  labels
-------------------------------- -  --------  -  -------------------------
108ef78ec381244447a108f931fe80db U            1  initialized, processed
be01a9fd6b3044cf12c4a83ee9612f84 U            2  initialized, processed
32764c28ef130baefebeba76a158ac4e U  process   3  initialized
# ...
>>>
```

Intro to Nanotribology

- Micro- and nano-electromechanical devices (MEMS and NEMS) used in a wide range of applications
  - Accelerometers
    - Airbag deployment
    - Consumer electronics
      - *Apple iPhone, Nintendo Wii*
  - Micromirror arrays
    - HD projectors (DLP TV)
  - Sensors
- Typically applications avoid parts in contact
  - Surface forces such as adhesion, stiction, friction, and wear dominate
- Lubrication schemes needed for devices to reach full potential

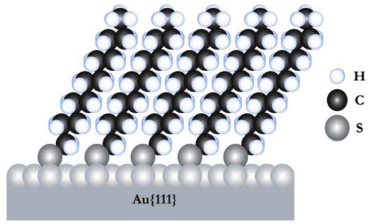


http://mems.sandia.gov/gallery/images/gc8.jpg

Monolayer Lubricants

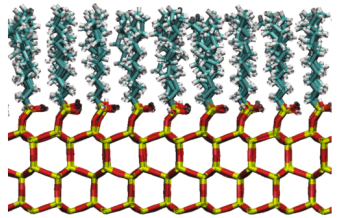
- Considerations for nanoscale lubricants
  - Low surface energy
    - **Minimize adhesion**
  - Low friction coefficient (μ)
    - **Minimize friction**
  - Robust/durable
    - **Wear-resistant**
- Alkylsilane monolayers
  - Densely-packed chains reduce surface energy, protect underlying surface from wear
    - Low adhesion between surfaces even in the presence of water
  - Friction coefficients vary widely by composition, density, and chain length
    - μ reduced to < 50% of value for bare silica[1]
  - Durability concerns

Alkanethiol monolayer on gold



Smith et al., *Prog. in Surf. Sci.*, 2004

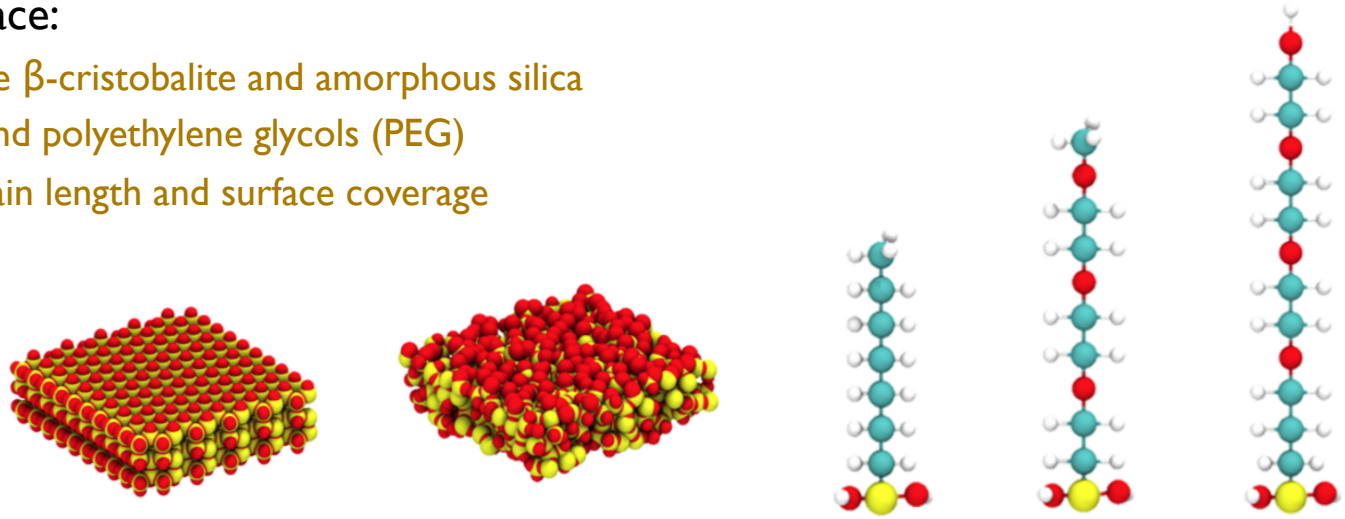
Alkylsilane monolayer on silica



Booth et al., *Langmuir*, 2011

Screening Monolayers (with metaMDS)

- Proposed hypotheses in the literature
  - More flexible chains reduce friction force on amorphous substrates
  - Crystalline monolayers imposed artificially induced ordering
    - Leads to lower friction coefficients
- Design space:
  - Crystalline β-cristobalite and amorphous silica
  - Alkanes and polyethylene glycols (PEG)
  - Varied chain length and surface coverage



Klein, C., Sallai, J., **Jones, T. J.**, Iacovella, C. R., McCabe, C. Cummings, P.T., "A Hierarchical, Component Based Approach to Screening Properties of Soft Matter," In Foundations of Molecular Modeling and Simulation: Select Papers from FOMMS 2015; R. Q. Snurr; C. S. Adjiman and D.A. Kofke, Ed.; Springer Singapore: Singapore, 2016; pp 79-92.

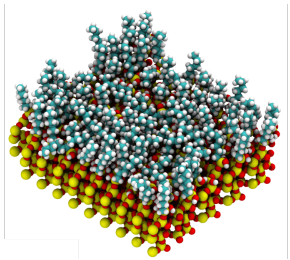
Nematic Order

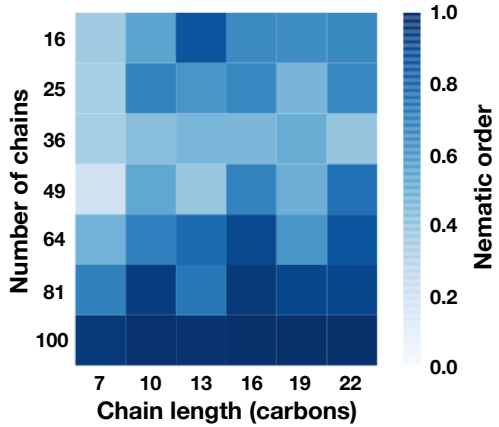
- **Nematic order** - measure of the global orientational ordering of monolayer chains
  - 0 = isotropic
  - 1 = perfectly ordered

$$S_2 = \langle \frac{3}{2}\cos^2\theta - \frac{1}{2} \rangle$$

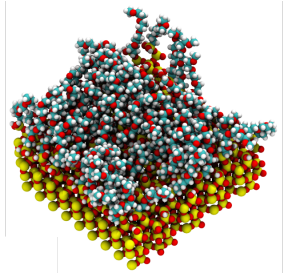
- PEG is found to feature lower nematic order than alkanes, particularly at lower surface coverage
  - Floppier/more liquid-like

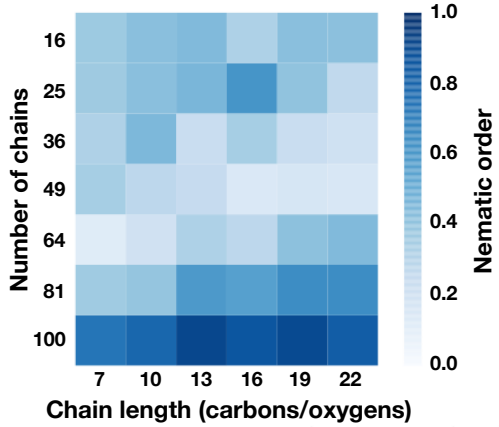
Alkanes





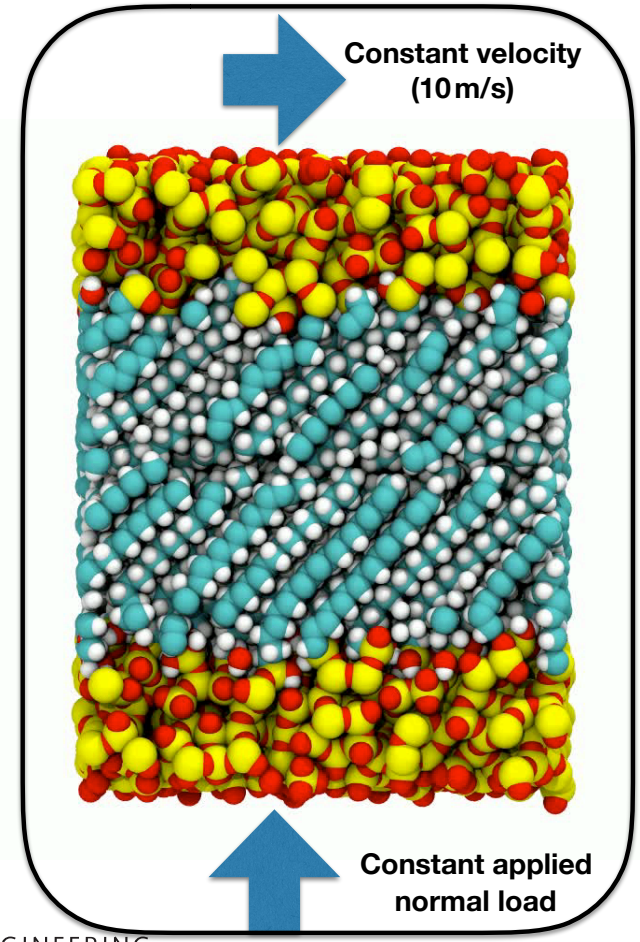
PEG





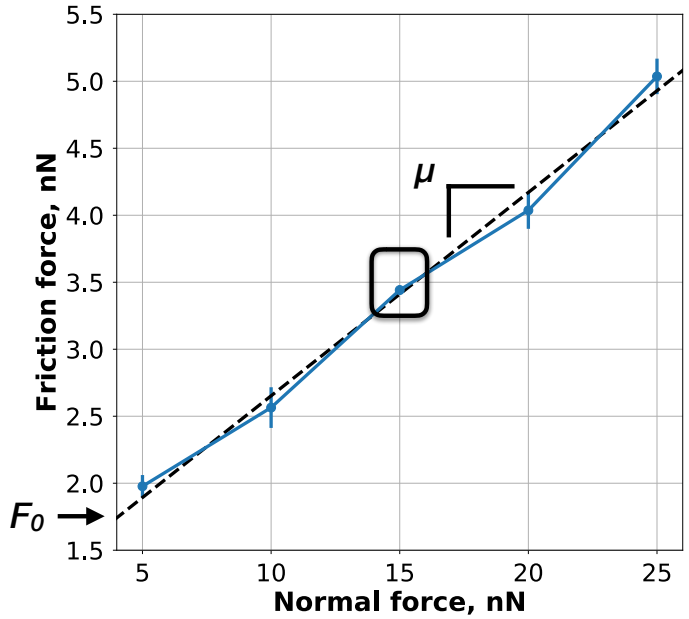
Klein, C., Sallai, J., **Jones, T. J.**, Iacovella, C. R., McCabe, C. Cummings, P.T., "A Hierarchical, Component Based Approach to Screening Properties of Soft Matter," In Foundations of Molecular Modeling and Simulation: Select Papers from FOMMS 2015; R. Q. Snurr; C. S. Adjiman and D.A. Kofke, Ed.; Springer Singapore: Singapore, 2016; pp 79-92.

Evaluating Lubricity

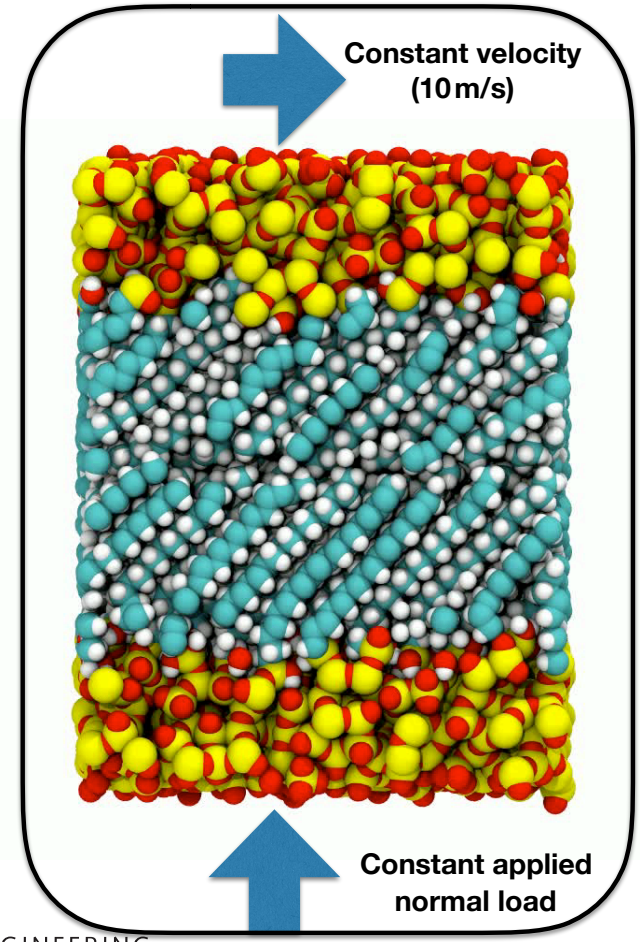


Derjaguin form of Amonton's Law of Friction

$$F_{\text{friction}} = F_0 + \mu F_{\text{normal}}$$

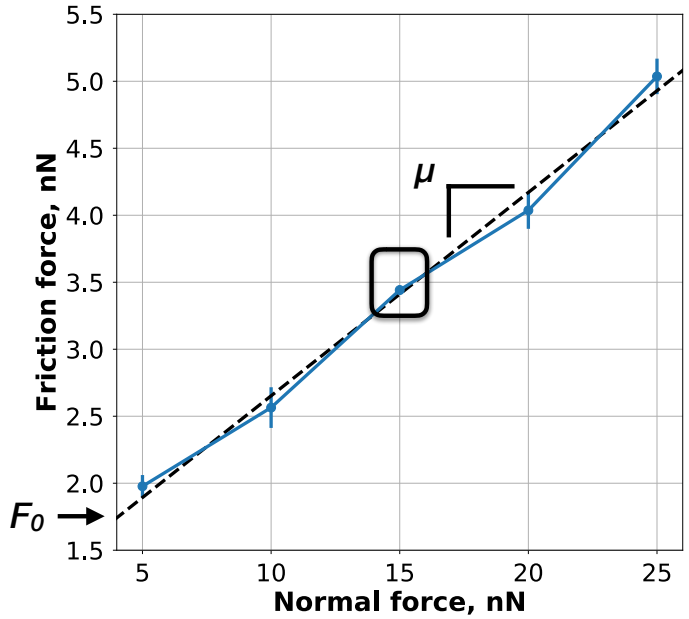


Evaluating Lubricity



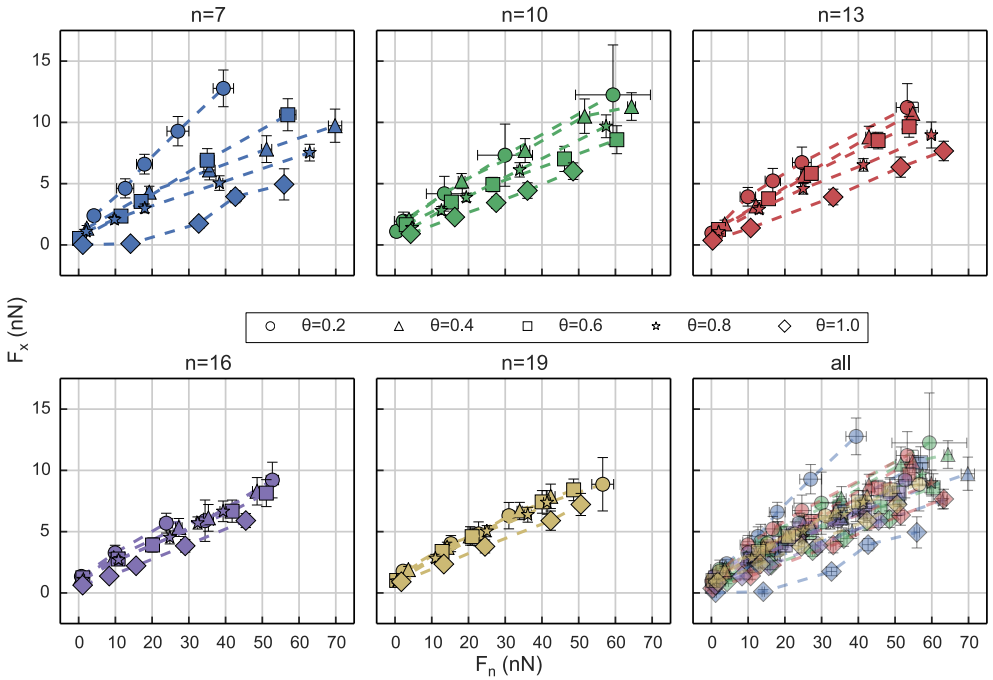
Derjaguin form of Amonton's Law of Friction

$$F_{\text{friction}} = F_0 + \mu F_{\text{normal}}$$



Surface Coverage has Greater Impact on Short Chains

- Subset of friction force data for polyethylene glycol monolayers on crystalline surface

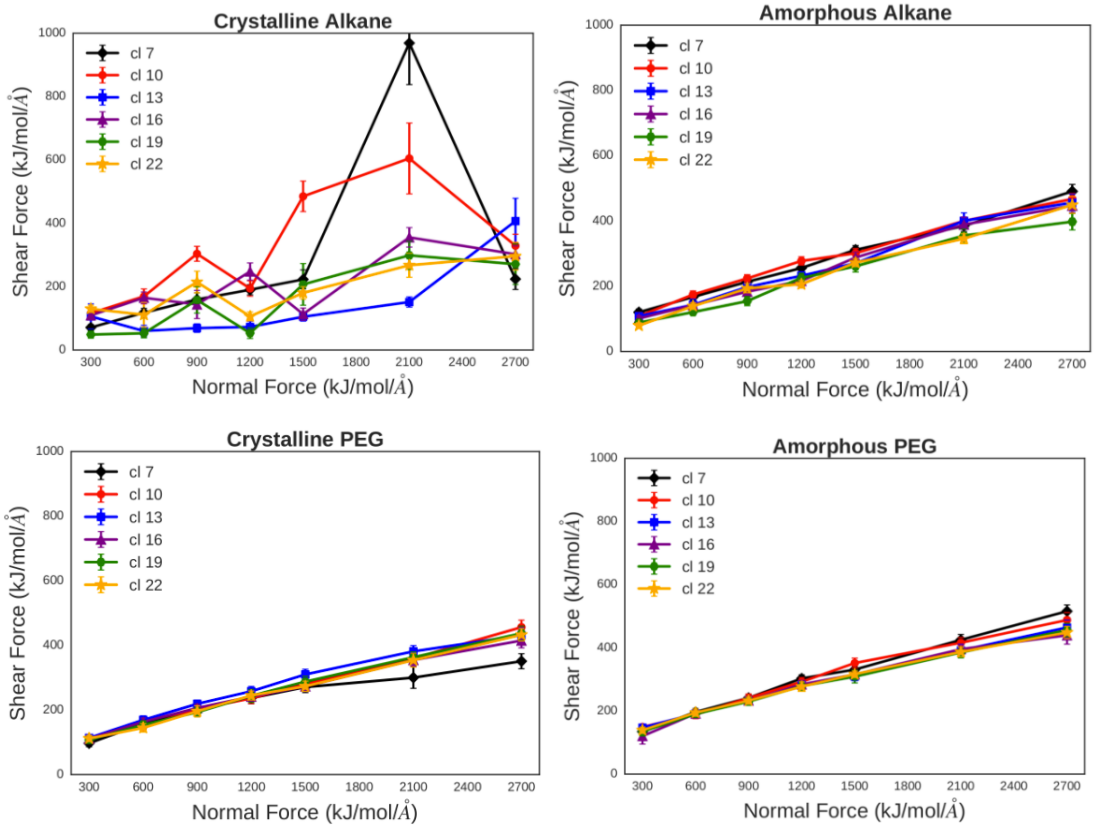


- n = number of C/O atoms in chain
- θ = fractional surface coverage

Klein, C., Sallai, J., **Jones, T. J.**, Iacovella, C. R., McCabe, C. Cummings, P.T., "A Hierarchical, Component Based Approach to Screening Properties of Soft Matter," In Foundations of Molecular Modeling and Simulation: Select Papers from FOMMS 2015; R. Q. Snurr; C. S. Adjiman and D.A. Kofke, Ed.; Springer Singapore: Singapore, 2016; pp 79-92.

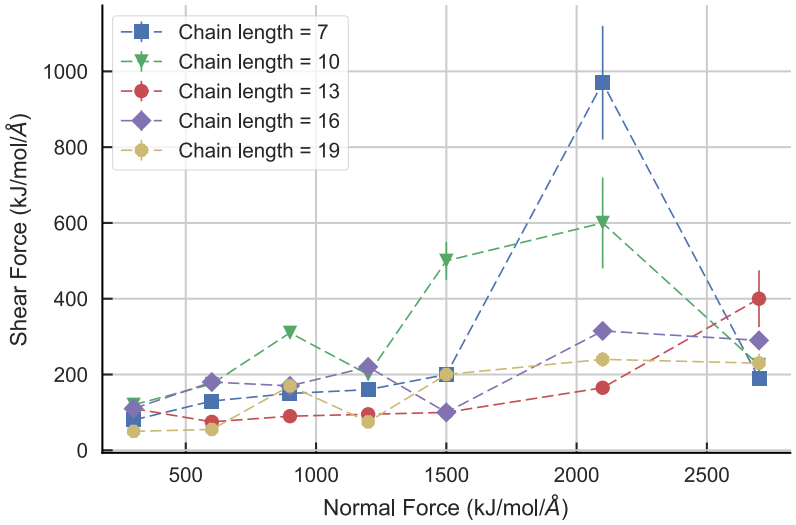
Friction Forces

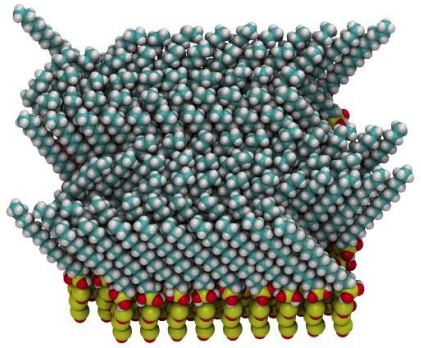
- Similar behavior for PEG and alkanes on amorphous surface
- Strange results for alkanes on crystalline surfaces



Crystalline Monolayers Can Exhibit artifacts

- Crystalline monolayers impose artificial ordering on monolayers
  - Portions of monolayer can get trapped in different configurations
  - High forces perpendicular to shear direction

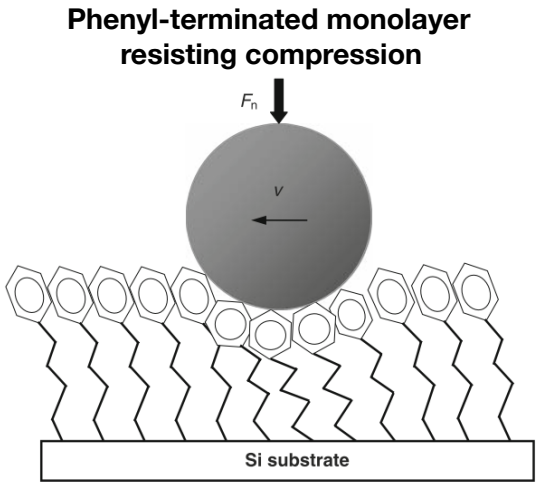




Klein, C., Sallai, J., **Jones, T. J.**, Iacovella, C. R., McCabe, C. Cummings, P.T., "A Hierarchical, Component Based Approach to Screening Properties of Soft Matter," In Foundations of Molecular Modeling and Simulation: Select Papers from FOMMS 2015; R. Q. Snurr; C. S. Adjiman and D.A. Kofke, Ed.; Springer Singapore: Singapore, 2016; pp 79-92.

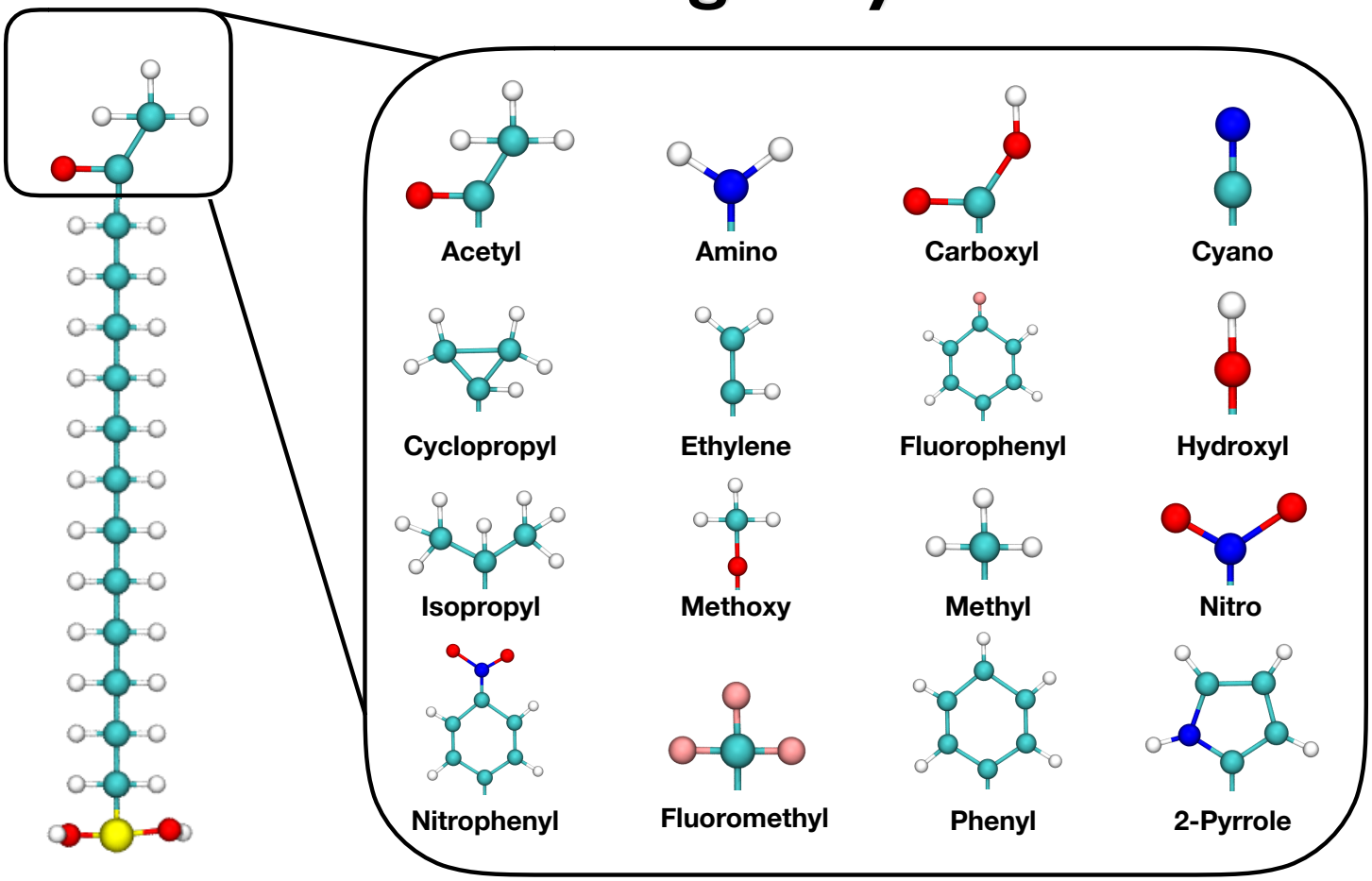
Terminal Group Chemistry

- Terminal groups form the boundary at the monolayer interface, thus having important effects on tribology
  - Methyl most common - low surface energy
- Higher friction forces in hydroxyl-terminated monolayers
  - Attributed to H-bonding across the monolayer interface
- Phenyl terminal groups can improve durability by resisting asperity penetration
- Only a few chemistries studied in the context of lubrication

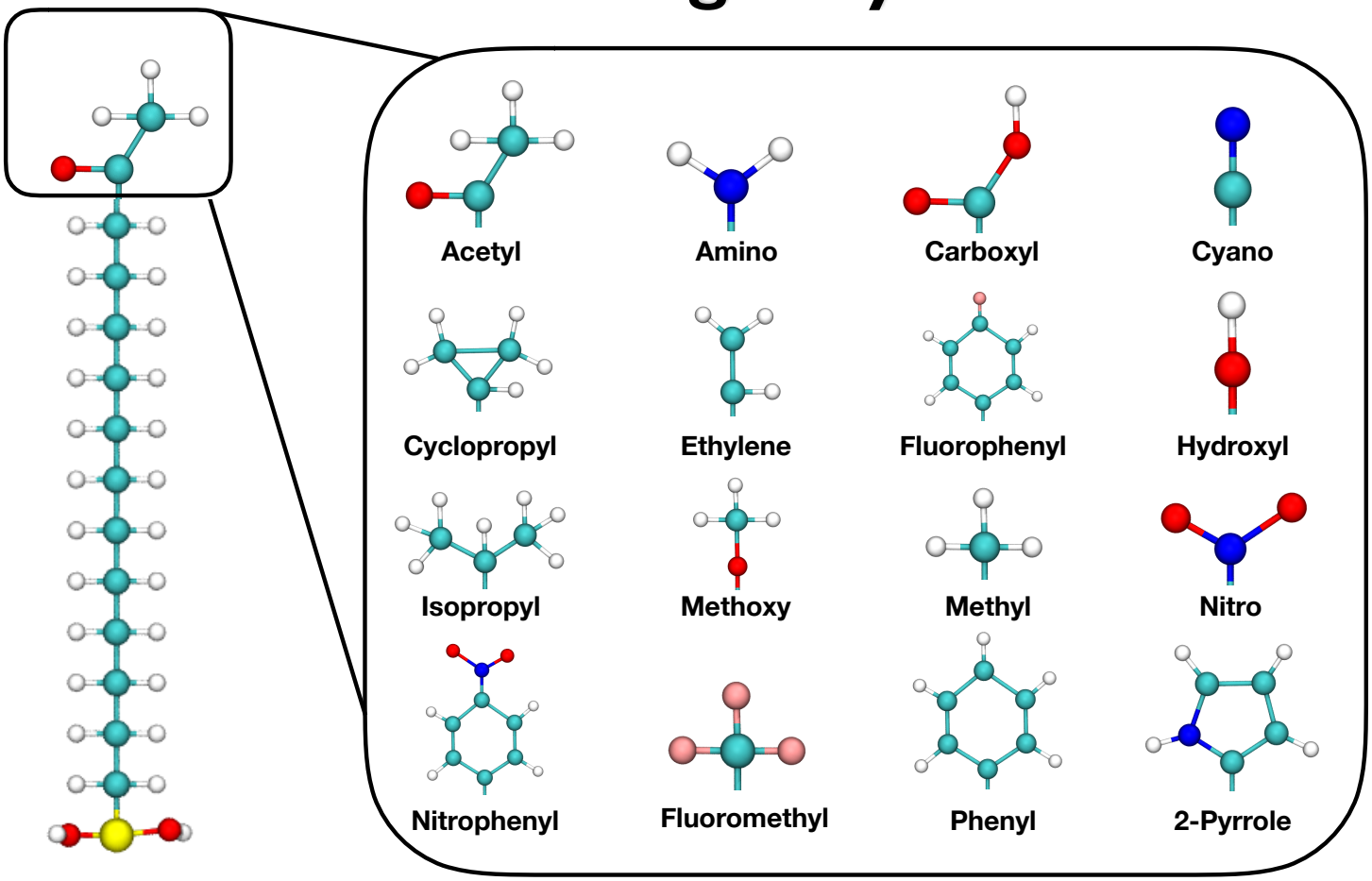


Yu et al., *Tribol. Lett.*, 2009

Constructing Alkylsilanes

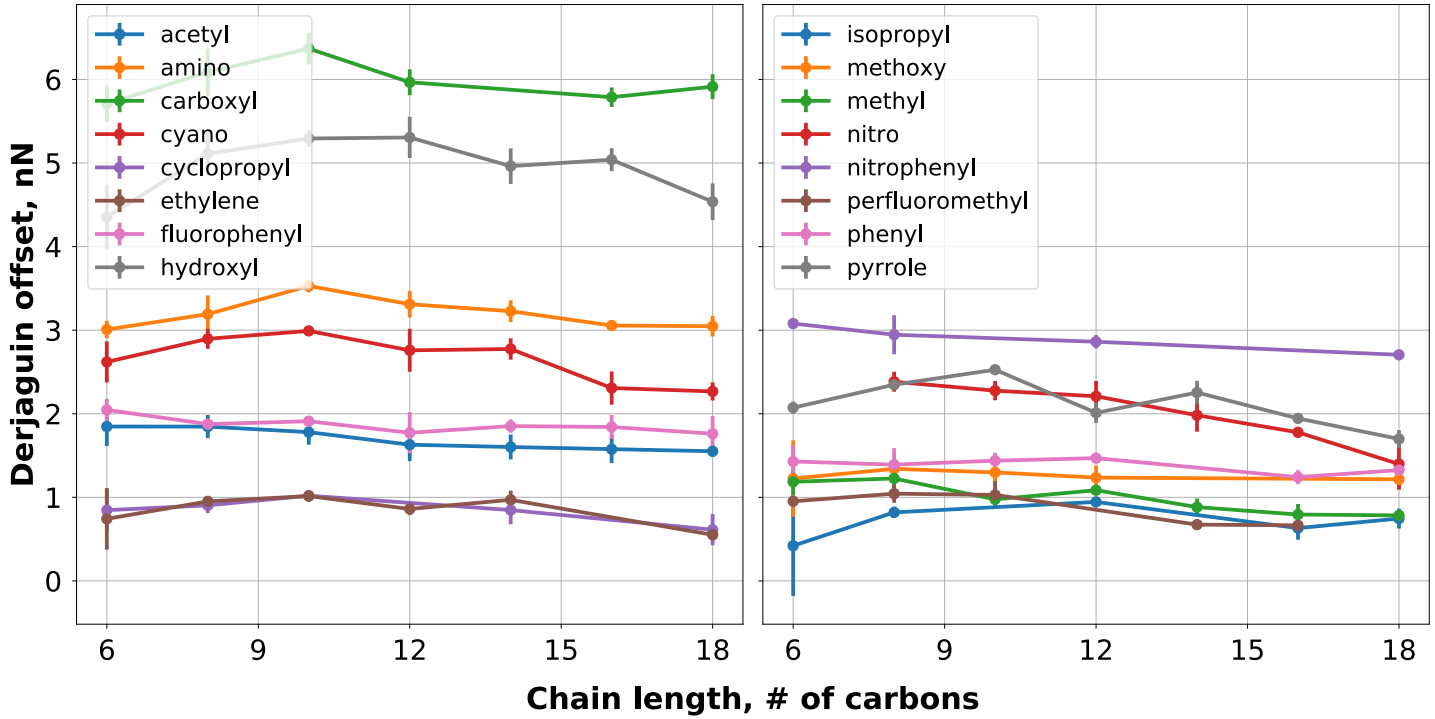


Constructing Alkylsilanes



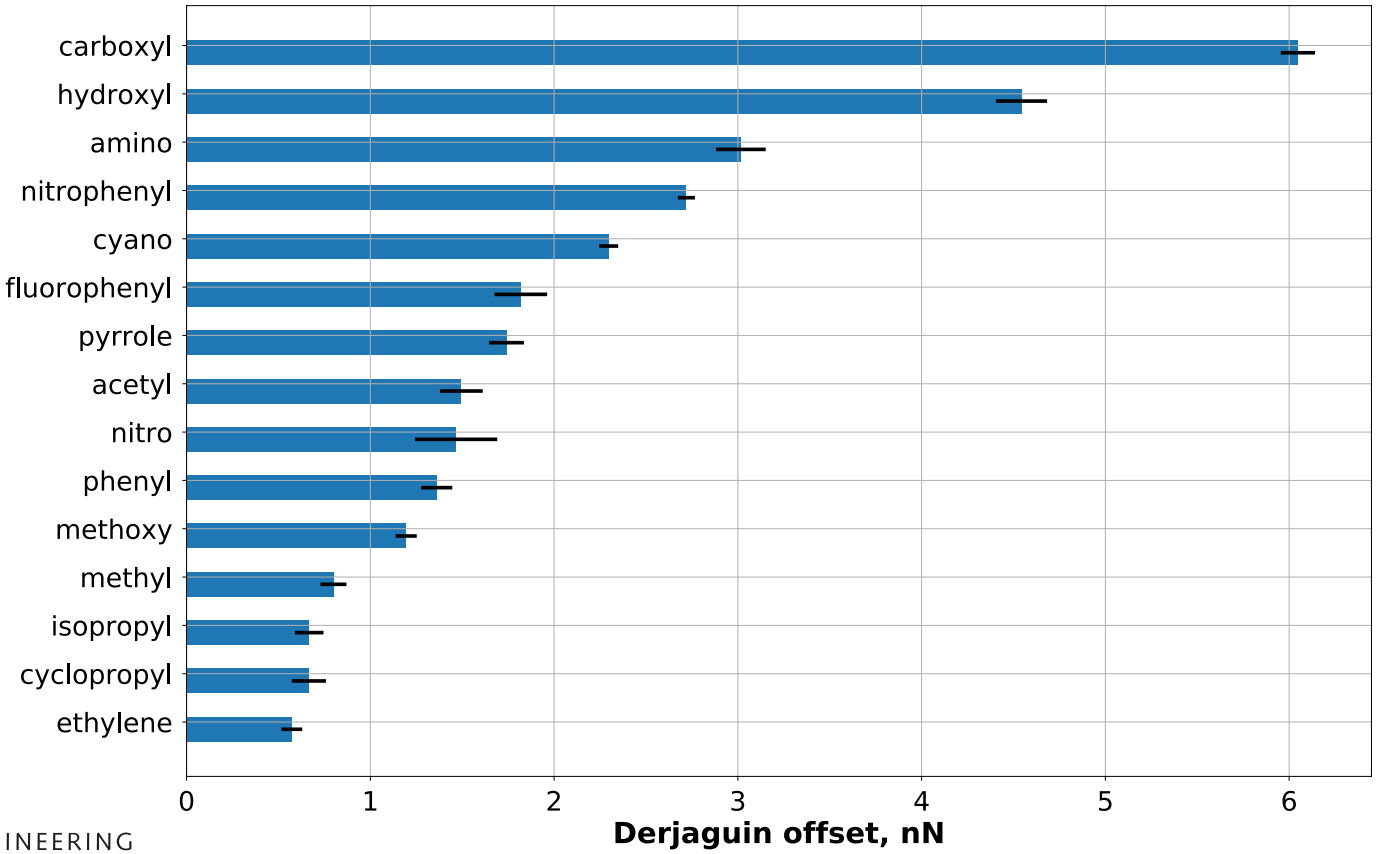
Derjaguin Offset

$$F_{\text{friction}} = \boxed{F_0} + \mu F_{\text{normal}}$$



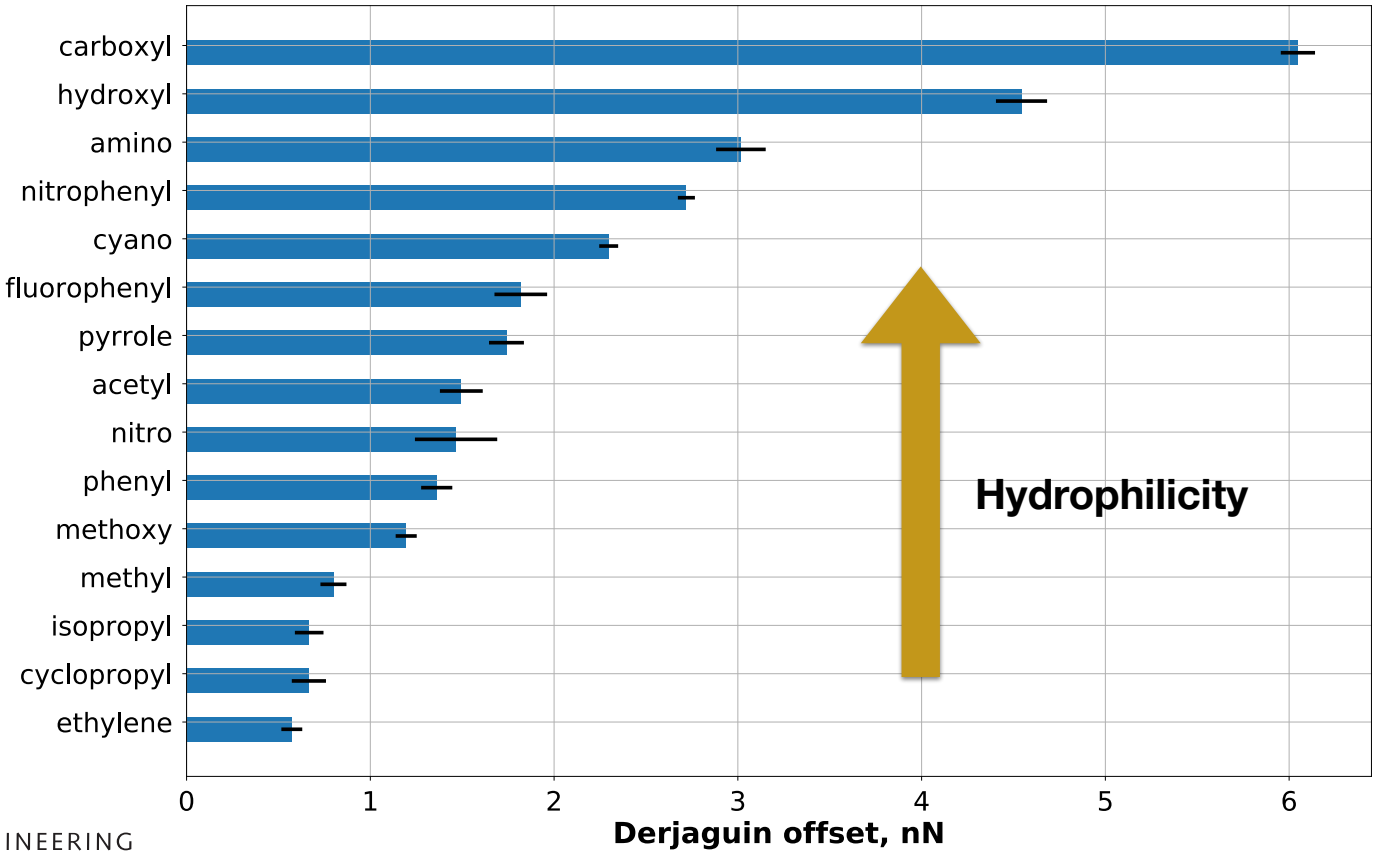
Derjaguin Offset (C18)

$$F_{\text{friction}} = \boxed{F_0} + \mu F_{\text{normal}}$$



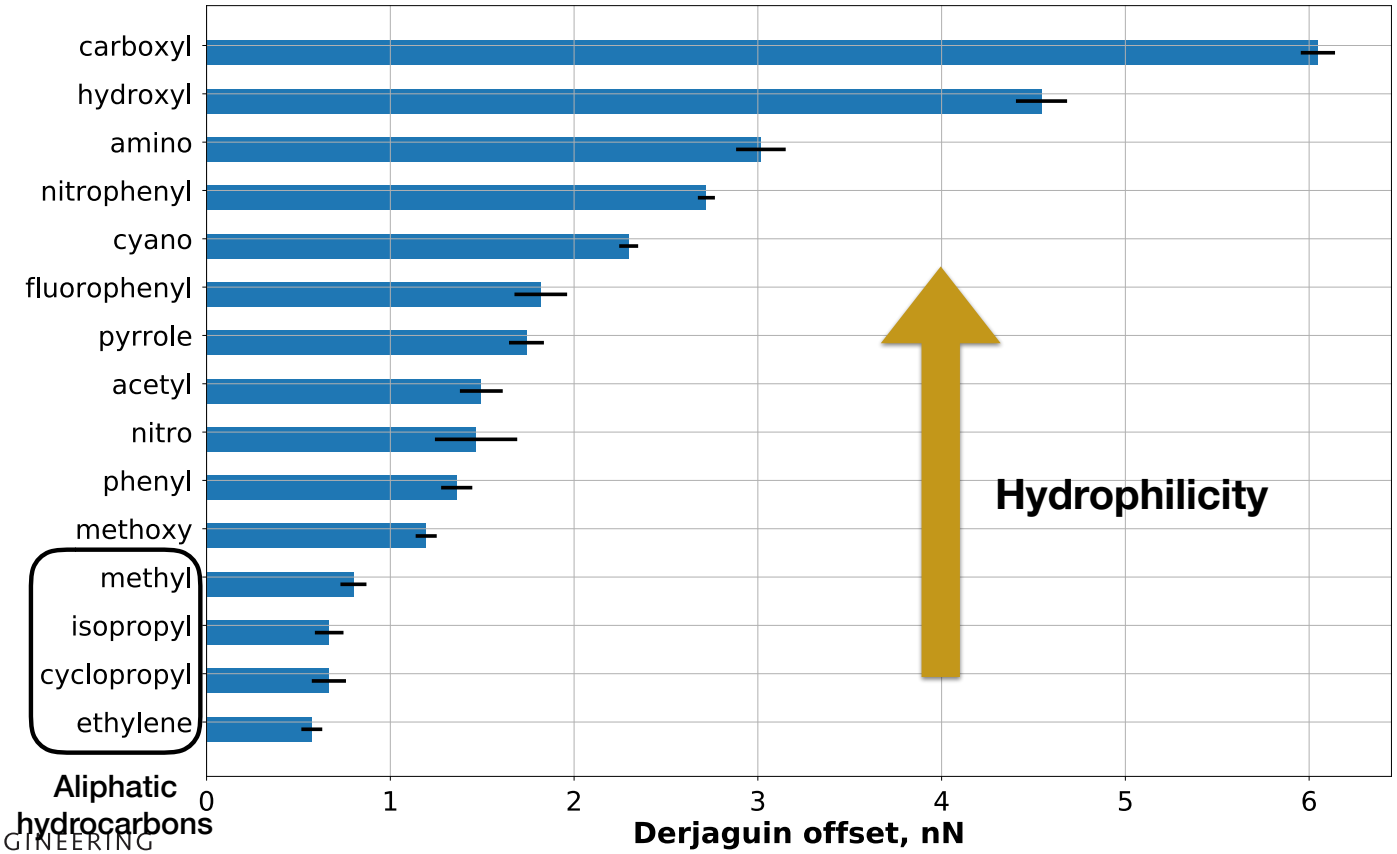
Derjaguin Offset (C18)

$$F_{\text{friction}} = \boxed{F_0} + \mu F_{\text{normal}}$$



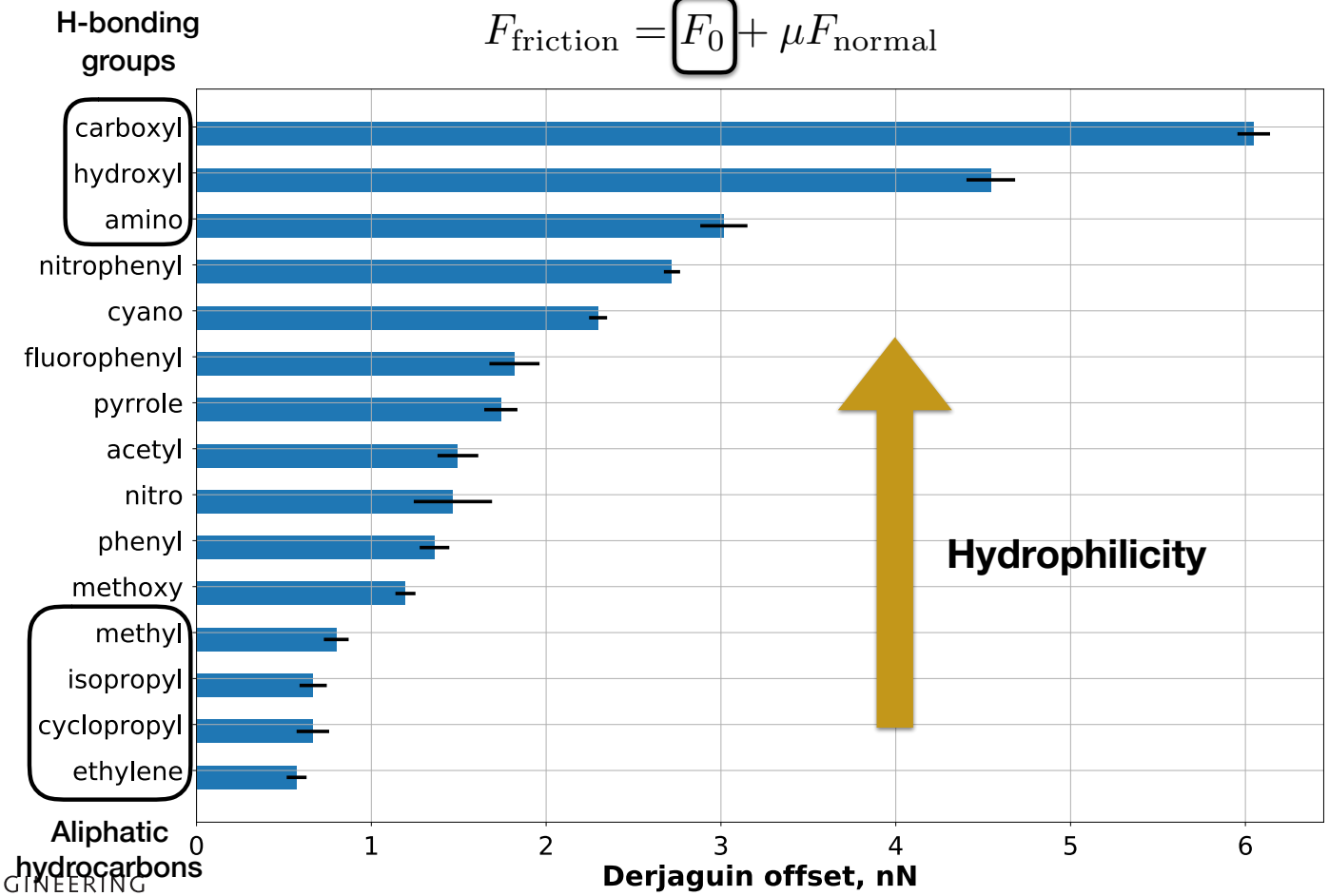
Derjaguin offset (C18)

$$F_{\text{friction}} = F_0 + \mu F_{\text{normal}}$$

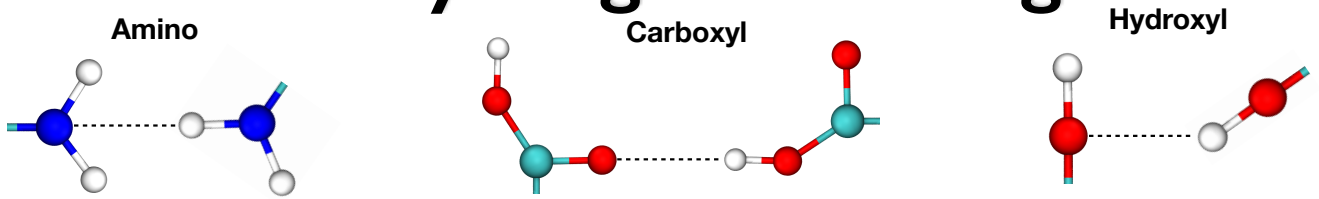


Derjaguin Offset (C18)

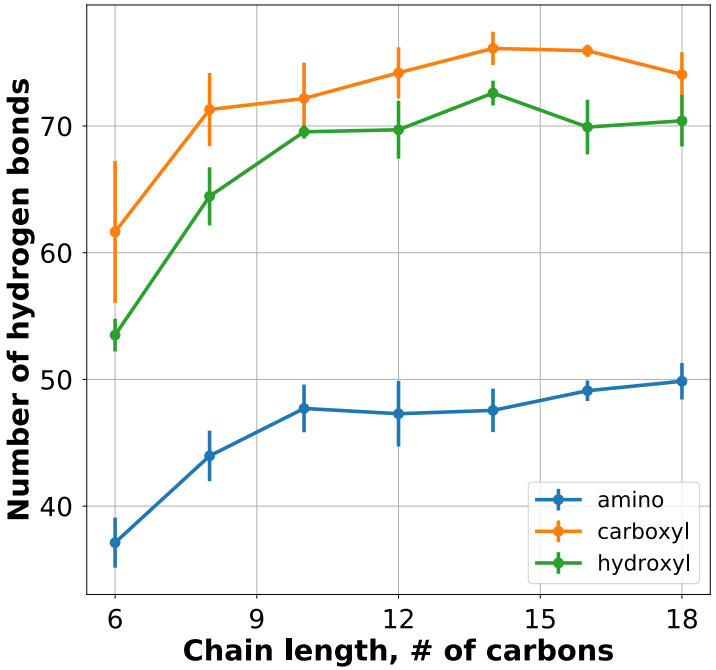
$$F_{\text{friction}} = F_0 + \mu F_{\text{normal}}$$



Hydrogen Bonding



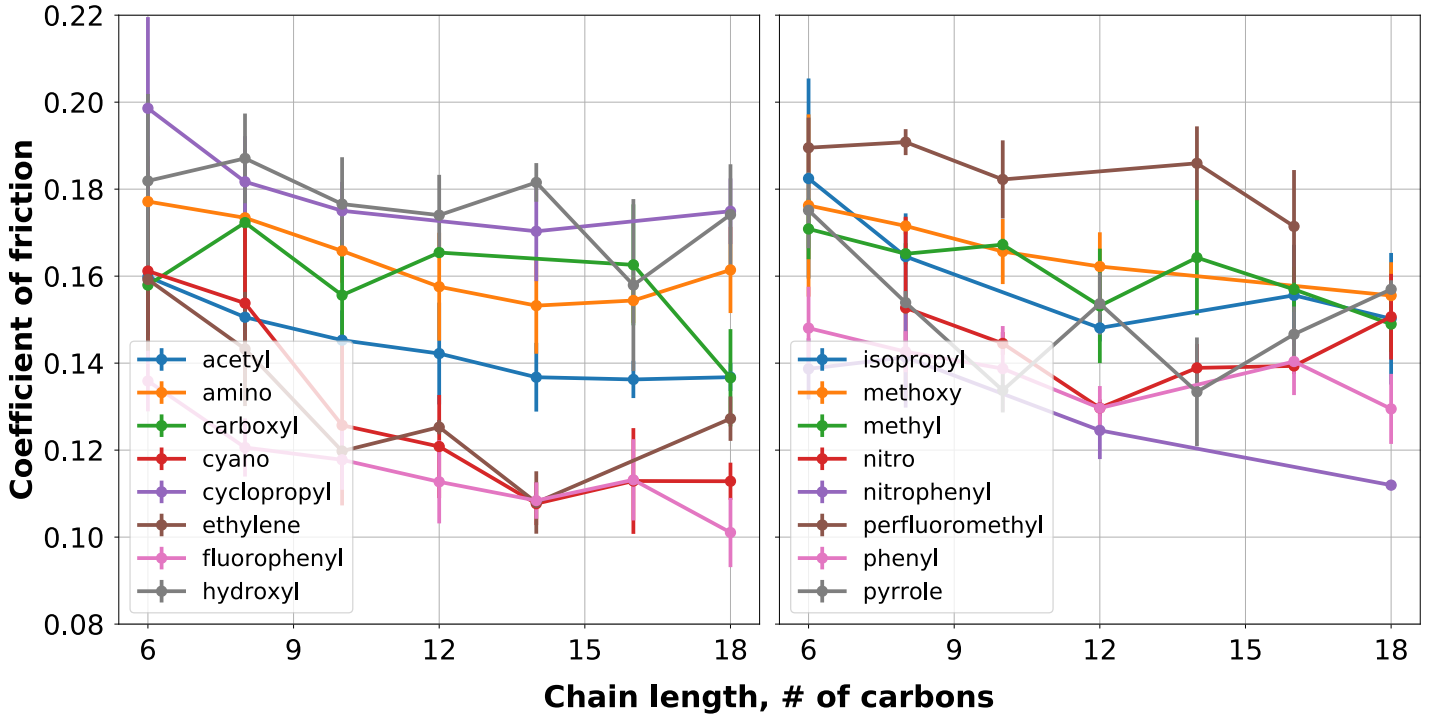
- $n_{amino} < n_{hydroxyl} < n_{carboxyl}$
  - Number of H-bonds determined using the Wernet-Nilsson method*
- Agrees with trends in Derjaguin offset - following expectations from the literature
- Number of H-bonds increases as backbone length increases from 6 to 10 carbons
  - Likely a result of increased orientational order of chains



*Wernet et al., *Science*, 2004

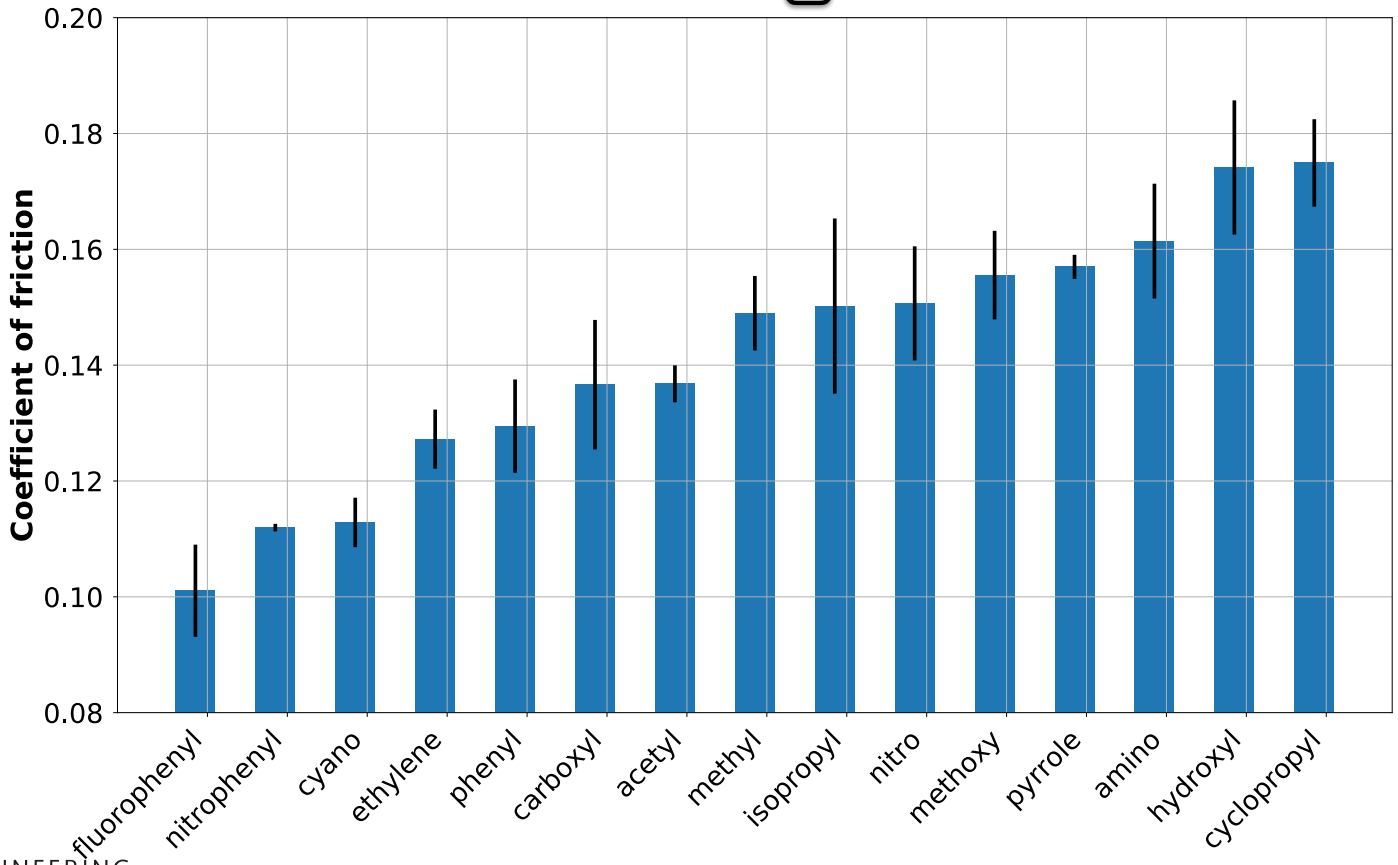
Coefficient of Friction

$$F_{\text{friction}} = F_0 + \mu F_{\text{normal}}$$



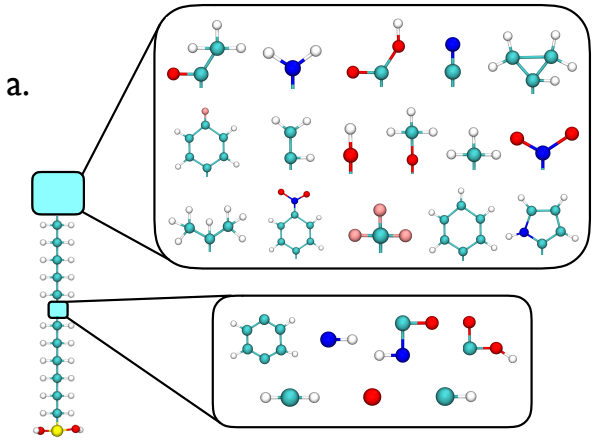
Coefficient of Friction - C18

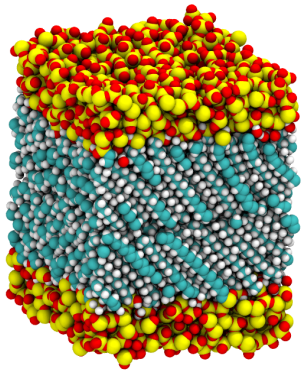
$$F_{\text{friction}} = F_0 + \mu F_{\text{normal}}$$



Next Steps

- Large parameter set screening
  - Other factors which affect tribological behavior
    - Different tether chemistry
    - Mixed monolayers
      - *Mixed length, end-functionalization, tether chemistry*
    - Asymmetric functionalization - different end groups on tethers
      - *For N ways to produce a monolayer, N(N+1)/2 asymmetric functionalization scenarios*
    - Identified ~45,000 variations to be screened
      - *Hope to perform on OLCF Titan in the next few months*





Conclusions

- To perform MGI-style screening of soft materials requires automating many steps currently largely performed by hand
- MoSDeF is an environment for automating molecular simulations
  - mBuild, Foyer, metaMDS,….
- Simulations performed within MoSDeF are completely reproducible
  - All steps are scripted, which can be distributed online through github
    - In line with identified best practices for reproducibility of computational research
- MoSDeF is being co-developed with other groups interested in transparency, reproducibility, usability, and extensibility (TRUE)
- Application to nanoscale lubrication is one example of application of soft materials screening made possible by MoSDeF and other tools (Signac)
- All resources available on github with version control
  - MoSDeF Hub - https://github.com/mosdef-hub
  - Foyer - https://github.com/mosdef-hub/foyer and mBuild - https://github.com/mosdef-hub/mbuild

Acknowledgments

- VU ChBE
  - Clare McCabe
  - Chris Iacovella
  - Christoph Klein
  - Andrew Summers
  - Justin Gilmer
  - Trevor Jones
- VU ISIS
  - Akos Ledezci
  - Gabor Karsai
  - Janos Sallai
  - Tengyu Ma

- Funding
  - NSF ACI 1047828: Development of an Integrated Molecular Design Environment for Lubrication Systems (iMoDELS)
  - NSF CDI CBET 1028374: Cyber-Enabled Design of Functional Nanomaterials
  - NSF SSE ACI 1535150: Development of a Software Framework for Formalizing Forcefield Atom-Typing for Molecular Simulation
- Resources
  - National Energy Research Supercomputing Center (NERSC), Lawrence Berekeley Laboratory
  - Oak Ridge Leadership Class (OLC), Oak Ridge National Laboratory