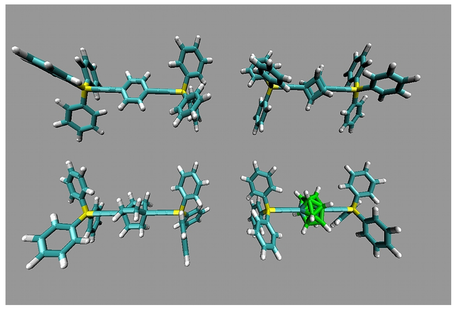
Using distributed computing to study crystalline molecular rotors.

Alexey V. Akimov
Department of Chemistry, Rice University, Houston, TX 77005, USA

Crystalline rotors

Objects studied:

Crystalline rotors with different rotators



Motivation:

To understand how rotor properties depend on symmetry and flexibility of rotators [1]

Theoretical method:

Molecular dynamics with rigid bodies = solution of set of differential equations of motion for some period of time

$$\dot{\vec{r}}_i = \vec{p}_i / m_i \qquad \dot{\vec{p}}_i = \vec{F}_i$$

$$\dot{\vec{l}}_i^{(e)} = \vec{\tau}_i^{(e)} + \vec{l}_i^{(e)} \times I^{-1} \vec{l}_i^{(e)}$$

$$\frac{d\mathbf{Q}}{dt} = \mathbf{Q}\, skew\, (\mathbf{I}^{-1}\boldsymbol{\pi})$$

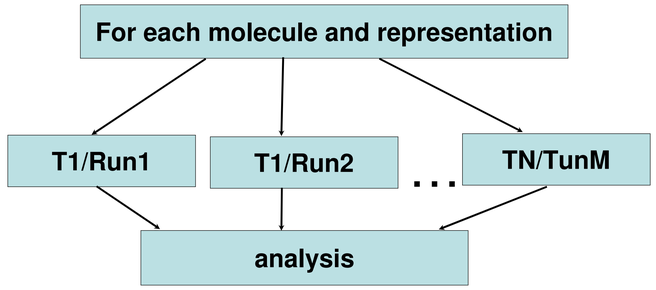
Challenges:

Many temperatures: T1, T2, T3, …
Many configurations: Run1, Run2, Run3, …
For each type of rotator: M1, M2, M3, …
Many representation: How many rigid groups
Need long trajectories
Periodic boundary conditions

Using local computing resources (SUG@R)

Shared University Grid at Rice (SUG@R)

134 SunFire x4150 nodes
each node: 2 quad-core Intel Xeon processors 2.83GHz
Max runtime policy: 24 hours



Simulation results [2]

Table 1. Summary of activation energies (kcal/mol) for different molecule representations and for different ways of accounting for 1,4 pair interactions

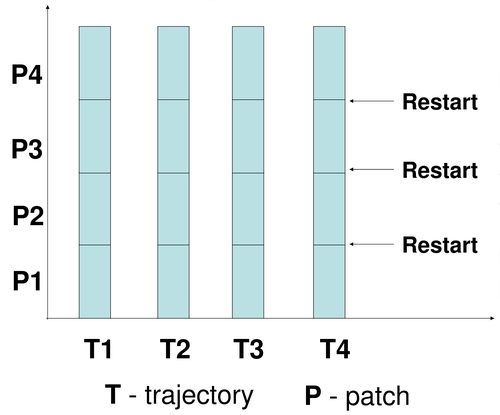
Representation\Molecule	Phe	BCO	Cub	Carbor
Experiment	8.5+/-2.5	3.5+/-0.2	12.6+/2.5	3.0+/-0.1
Flex with 1,4 pairs	8.0+/-0.8	3.7+/-0.3	6.6+/-0.9	5.0+/-0.7
Flex without 1,4 pairs	8.0+/-0.7	4.9+/-0.5	6.3+/-0.9	4.9+/-0.4
Flex1 with 1,4 pairs	4.3+/-0.5	3.9+/-0.3	6.3+/-0.7	5.3+/-0.5
Flex1 without 1,4 pairs	5.8+/-0.5	4.5+/-0.5	5.8+/-0.8	3.5+/-0.6
Flex2 with 1,4 pairs	5.5+/-0.6	3.1+/-0.4	5.1+/-0.4	6.1+/-1.0
Flex2 without 1,4 pairs	5.8+/-0.5	3.8+/-0.8	3.4+/-0.5	3.0+/-0.5
Rod with 1,4 pairs	5.1+/-0.6	3.9+/-0.5	4.4+/-0.4	5.5+/-0.6
Rod without 1,4 pairs	4.9+/-1.4	2.6+/-0.3	3.3+/-0.7	2.5 +/- 0.4

We did	We need
Typically 10-20 temperatures	~ 20 temperatures
Only 5 initial configurations	~100 initial configurations or more
4 molecules	~10 new molecules or more
4 representations	~5-10 representations or more
1-4 ns trajectories	~10-100 ns trajectories
No electrostatics	With electrostatics

Using distributed computing resources (OSG)

Workflow for longer trajectories, 2 key-components:

Patching trajectories



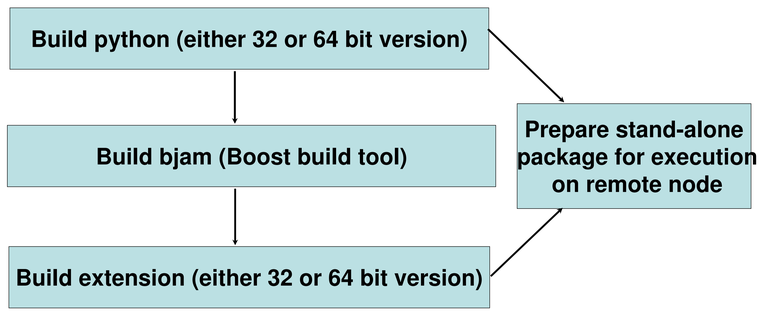
Using DAGMan:

For each trajectory:

JOB t1p1 t1p1.sub
JOB t1p2 t1p2.sub
JOB t1p3 t1p3.sub
…
PARENT t1p1 CHILD t1p2
PARENT t1p2 CHILD t1p3
…

Binary preparation:

python extension, depends on Boost libraries
must be compiled with correct python interpreter
using correct address model for worker nodes



References

[1] Karlen, S. D.; Reyes, H.; Taylor, R. E.; Khan, S. I.; Hawthorne, M. F.; Garcia-Garibay, M. A. *Proc. Natl. Acad. Sci. USA* **2010**, *107*, 14973
[2] Akimov, A. V.; Kolomeisky, A. B. *J. Phys. Chem. C* **2011**, 115, 13584

Acknowledgments

- Prof. Anatoly.B. Kolomeisky, Rice University, Chemistry (Ph. D. adviser)
- Dr. Alain. Roy, University of Wisconsin-Madison, Computer Science (OSG mentor)
- OSG Summer School organizers and lectors
- Shared University Grid at Rice (SUG@R)