PSI4 workshop
Nov. 2014

Rollin A. King

Professor of Chemistry
Bethel University
St. Paul, Minnesota

Guest Professor
ETH Zürich

optking status

2011	P-RFO for transition states (Jon Cayton) IRC-following (Andreas Copan)
2012	steepest-decent (Tony Burand)
2013	rigid-body optimizations (Andreas Copan; Tony Burand)
spring 2014	restricted-step RFO (Sarah Elliott)
Fall 2014	Common simple coordinate base class. Separation of intramolecular and intermolecular coordinates. Cartesian and combination (delocalized). Increased robustness for problematic cases. Integrate distinct treatment for intermolecular coordinates (for flexible, and fixed-bodies)

Total number of steps to optimize the 30-molecule JCC93 test set with various Hessian guesses.

STO-3G			6-31G*		
H Guess	**# steps**		**H Guess**	**# steps**	**# fails**
Lindh simple	187		**Schlegel-1984**	**190**	0
Schlegel-1984	**190**		Fischer-1992	223	0
Fischer-1992	211		Lindh simple	231	0
Lindh once	249		Thumb	295	0
Thumb	264		Lindh once	268	1
Lindh all	343		Lindh all	314	1

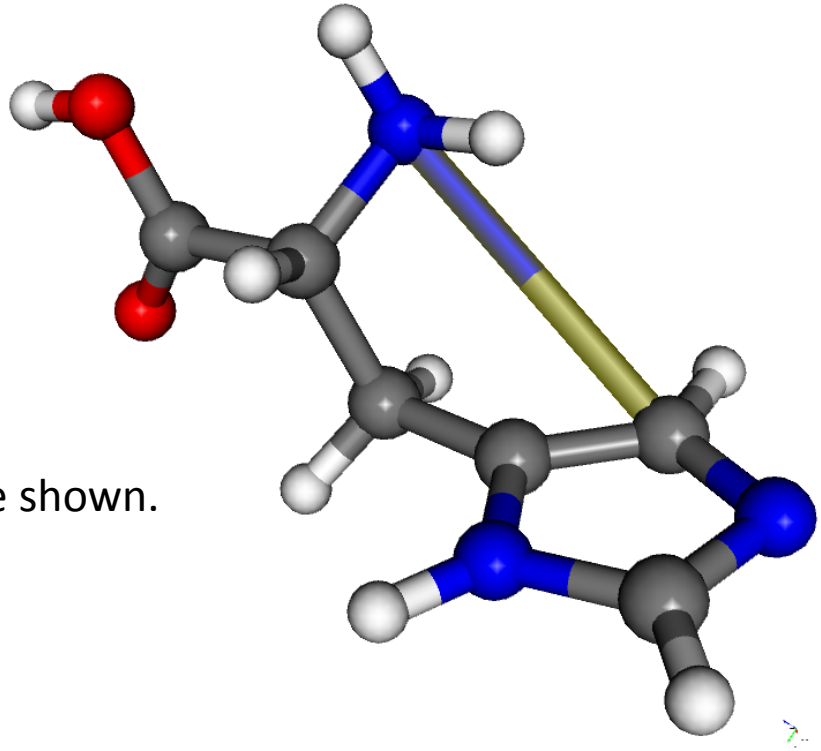
J. Baker, *J. Comput. Chem.* **14** 1085 (1993).
T.H. Fischer and J. Almlöf, *J. Phys. Chem.* **96** 9768 (1992).
R. Lindh, *Chem. Phys. Lett.* **241** 423 (1995).
H.B. Schlegel, *Theoret. Chem. Acta (Berl.)* **66** 333 (1984).

Total number of steps to optimize the full (30-molecule) JCC93 test set with various Hessian updates.

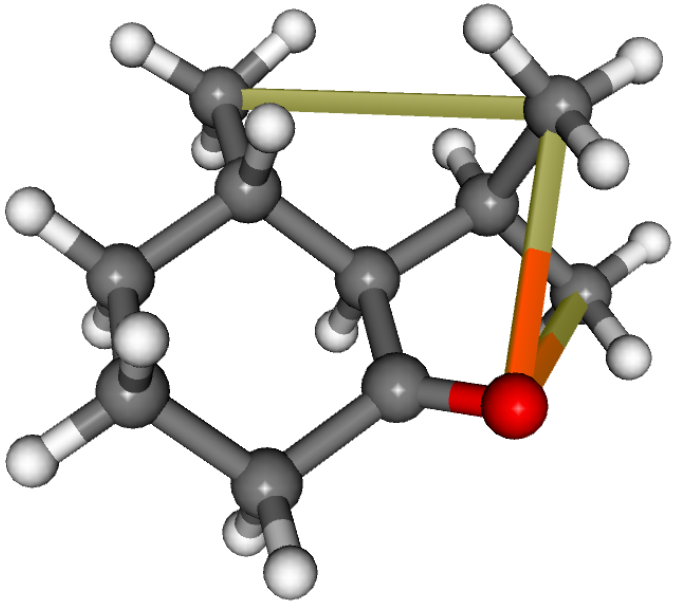
H Update	# steps		Use Previous Points	# steps
BFGS	**190**		1	215
Bofill	222		**2**	**209**
Murtagh/Sargent	234		3	226
Powell	250		4	208
None	362		5	214
			6	235

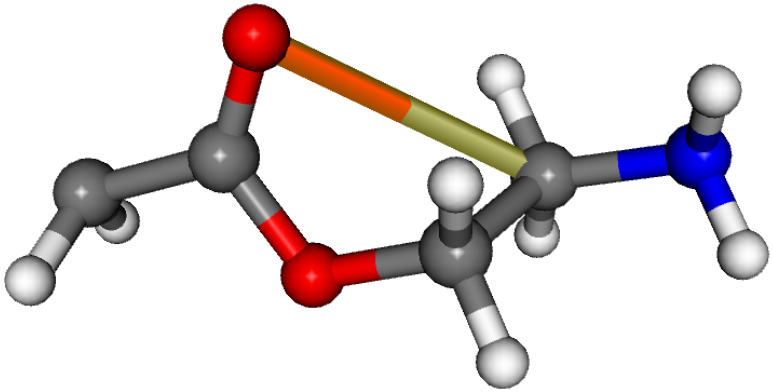
See J. M. Bofill, *J. Comp. Chem.* **15** 1 (1994).

Found 1-5 auxiliary bond coordinates speed convergence



Examples from JCC93 test suite shown.





JCC93 performance comparison	STO-3G	6-31G*	
Baker-1993	240		
Lindh-1995	215		
Eckert-1997	196		
Helgaker-2002	**185**	**198**	best previous performance
	174		exact initial hessians
	115		exact Hessians throughout
optking-2014 default	**182**	**174**	
optking-2014 dynamic_level=1	**196**	**189**	6-31G* includes 3 backsteps+ 2 level raises

JCC93 successful with various coordinate types

	# steps	# fails
redundant	188	0
delocalized	213	0
cartesian	304	0
both	327	0

P. Pulay and G. Fogarasi, *J. Chem. Phys.* **96** 2856 (1992).
C. Peng et. al, *J. Comp. Chem.* **17** 49 (1996).
J. Baker and W.J. Hehre, *J. Comp. Chem.* **12** 606 (1991).

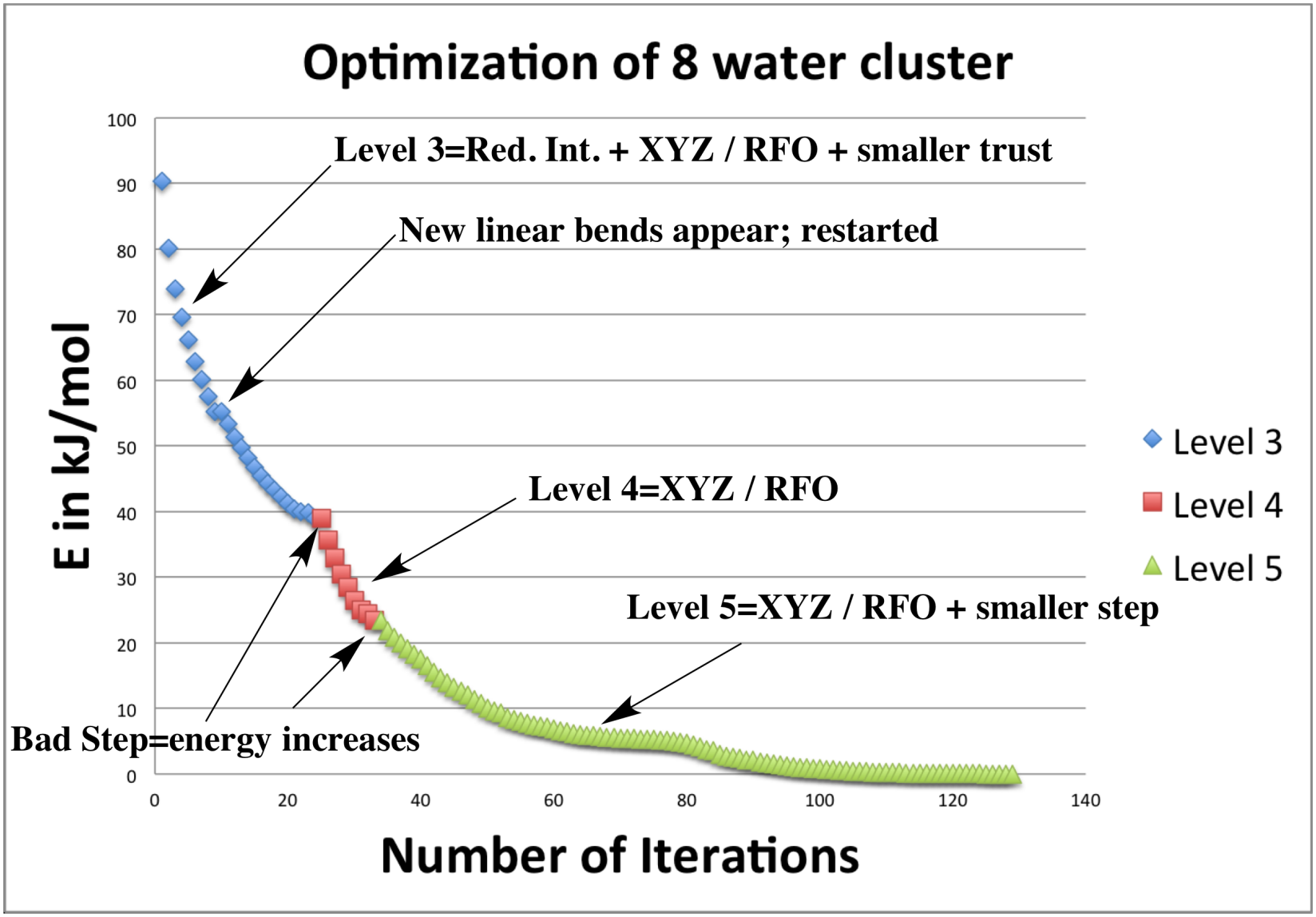
STO-3G HF

Dynamic optimization levels

level	step	coord-inates	step limit	backsteps if bad step?
1	RFO	RI	dynamic	no
2	RFO	RI	medium	yes (1)
3	RFO	RI+XYZ	small	yes (1)
4	RFO	XYZ	medium	yes (1)
5	RFO	XYZ	small	yes (1)
6	SD	XYZ	medium	yes(1)
7	SD	XYZ	small	yes(1)
	abort			

Default algorithm totally fails (can't converge backtransformation)!

Level 1=Red. Int. / RFO
Level 2=Red. Int. / RFO + backsteps + smaller trust.



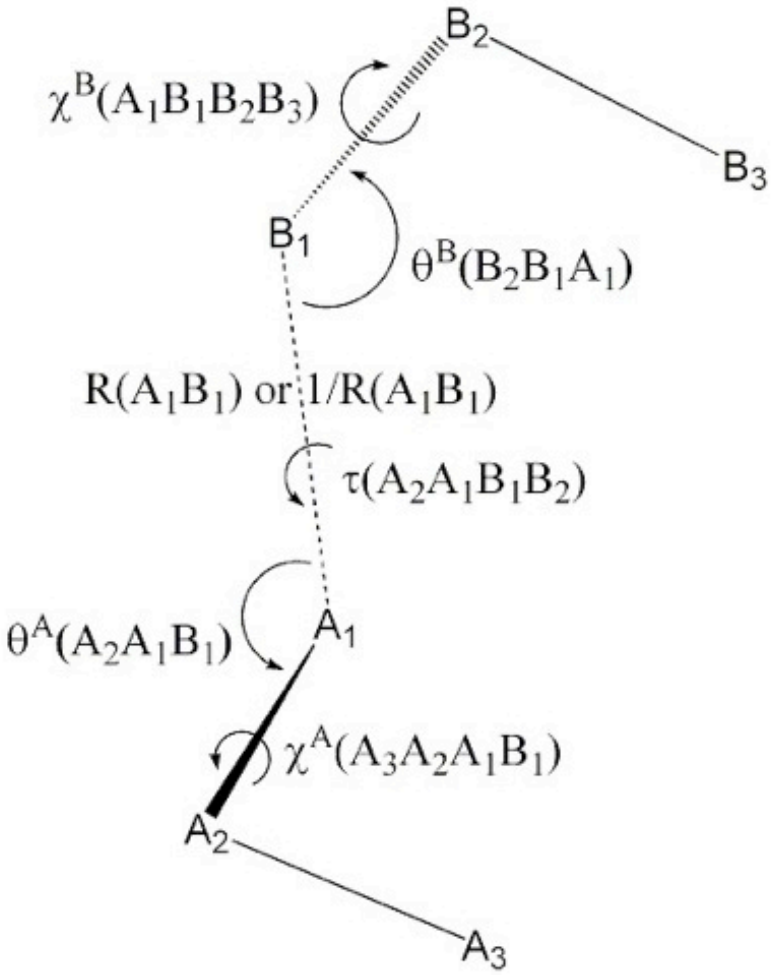
Supporting interfragment coordinates

1. user specifies linear combination of atom positions that define up to 3 reference points on each fragment, then the coordinates values that relate these reference points on different fragments. [My issue #1: NEED INPUT USER MECHANISM, lost in psi3->psi4].
2. we can define the reference points based on the fragment principal axes (the code is still in development; but I am worried about the consistency of the diagonalization of the inertia tensor from step to step).
3. need COM plus Euler angle coordinates for EFP or other fixed-body fragments. [My issue #2: NEED INPUT USER MECHANISM]

Interfragment Coordinates

Reference points may be linear combination of atoms or determined by principal axes.

Displacements can be done analytically – no back-transformation to Dx.



W. D. Allen, Univ. of Georgia

XML/CML with H.P. Lüthi (ETH Zurich)

Our objective: structured, machine-queryable output for study of molecular properties.

a) Use xml databases and tools.
b) Avoid text output parsers.
c) Use “common ontology” where available.
d) Not focused on roundtrip workflows.
e) 100’s to 1000’s of computations.

"The semantics of Chemical Markup language (CML) for computational chemistry: CompChem"

Positives:

- well-defined convention/dictionary online
- reasonable structure for quantum chemistry
- designed by those who know CML
- working validator code

Phadungsukanan, Kraft, Townsend, Murray-Rust,
Journal of Cheminformatics **2012**, 4:15.

I. jobList

A. job

1. initialization

a. up to 1 molecule

b. up to 1 parameterList

2. calculation

a. up to 1 molecule

b. up to 1 parameterList

c. up to 1 propertyList

d. additional calculations

3. finalization

a. up to 1 molecule

b. up to 1 propertyList

4. environment

a. up to 1 propertyList

CompChem convention structure

CompChem Negatives:

- demonstration involved only reading G03 (in 2012).
- tentative basis-set and array definitions.
- commitment level?
  - http://validator.xml-cml.org "not found"
  - http://www.xml-cml.org/dictionary/compchem/gaussian "not found"
  - incomplete 2011 convention online unmodified.
  - has anyone adopted fully?

"From data to analysis: linking NWChem and Avogadro with the syntax and semantics of CML"

Positives:

- produces CML output directly from NWChem (HPL's group can do this with Turbomole)
- new definition in CML for molecular orbitals
- integration with Avogadro for visualizing orbitals/animating vibrations

de Jong, Walker, Hanwell, *J. of Cheminformatics* **2013**, 5:25.

"From data to analysis: linking NWChem and Avogadro with the syntax and semantics of CML"

Negatives:

- Does not follow CompChem convention completely
- Publicly available Avogadro doesn't work (v.2 coming)
- Changed basis-set convention from CompChem/EMSL.
- very simple examples given (e.g., scf energy)
- Did not process input options into parameters.
- external FoX library necessary to collect data, produce the xml.

PSI4 XML

1. PSI4's top-level python interface is superior!

2. RAK has constructed a python module xml_psi that mimics the structure of CompChem.

3. xml_psi_jobList's can be loaded into an ElementTree (standard python library) object for xml output.

4. alternative output formats can be supported.

What works now

```
import xml_psi

### jobList
my_calc = xml_psi.xml_psi_jobList("PSI4 joblist title”)

### job
my_calc.add_job("optimization”)

h2o.update_geometry()

### job initialization
my_calc.jobs[0].initialization.parameters.add_string("psi:basis", "cc-pvtz")

my_calc.jobs[0].initialization.molecule.set_label("start_geom")
for i in range(h2o.natom()):
  my_calc.jobs[0].initialization.molecule.add_atom(h2o.symbol(i), h2o.x(i), h2o.y(i), h2o.z(i))
```

```
### job calculation

optimize('scf')

### job finalization

my_calc.jobs[0].finalization.molecule.set_label("end_geom")
for i in range(h2o.natom()):
  my_calc.jobs[0].finalization.molecule.add_atom(h2o.symbol(i), h2o.x(i), h2o.y(i), h2o.z(i))

### job environment
my_calc.jobs[0].environment.properties.add_string("cc:program",        "psi")
my_calc.jobs[0].environment.properties.add_string("cc:programVersion", version())
my_calc.jobs[0].environment.properties.add_string("cc:hostName", socket.gethostname())
```

rking@gimli:~/chem — vim — 125×55

```
<?xml version='1.0' encoding='utf-8'?>
<module convention="convention:compchem" xmlns:="http://www.xml-cml.org/schema" xmlns:cc="http://www.xml-cml.org/dictionary/c
ompchem/" xmlns:cml="http://www.xml-cml.org/schema" xmlns:compchem="http://www.xml-cml.org/dictionary/compchem/" xmlns:conven
tion="http://www.xml-cml.org/convention/" xmlns:conventions="http://www.xml-cml.org/convention/" xmlns:dummy="file:///Users/r
king/lib/python/dictionary/psi.xml" xmlns:h="http://www.w3.org/1999/xhtml" xmlns:n="http://www.xml-cml.org/dictionary/nwchem/
" xmlns:nonsi="http://www.xml-cml.org/unit/nonSi/" xmlns:si="http://www.xml-cml.org/unit/si/" xmlns:x="http://www.xml-cml.org
/dictionary/cmlx/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <module dictRef="compchem:jobList" title="PSI4 joblist title">
    <module dictRef="compchem:job" title="optimization">
      <module dictRef="compchem:initialization">
        <molecule convention="convention:molecular" formalCharge="0" id="start_geom" spinMultiplicity="1">
          <atomArray>
            <atom elementType="H" id="a1" x3="0.0000000000" y3="1.5479730249" z3="-0.9625975834" />
            <atom elementType="O" id="a2" x3="0.0000000000" y3="0.0000000000" z3="0.1213047976" />
            <atom elementType="H" id="a3" x3="0.0000000000" y3="-1.5479730249" z3="-0.9625975834" />
          </atomArray>
        </molecule>
        <parameterList>
          <parameter dictRef="psi:basis">
            <scalar dataType="xsd:string">cc-pvtz</scalar>
          </parameter>
        </parameterList>
      </module>
      <module dictRef="compchem:finalization">
        <molecule convention="convention:molecular" formalCharge="0" id="end_geom" spinMultiplicity="1">
          <atomArray>
            <atom elementType="H" id="a1" x3="0.0000000000" y3="1.4192337607" z3="-0.9497464864" />
            <atom elementType="O" id="a2" x3="0.0000000000" y3="0.0000000000" z3="0.1196853257" />
            <atom elementType="H" id="a3" x3="0.0000000000" y3="-1.4192337607" z3="-0.9497464864" />
          </atomArray>
        </molecule>
      </module>
      <module dictRef="compchem:environment">
        <propertyList>
          <property dictRef="psi::process_id">
            <scalar dataType="xsd:string">45758</scalar>
          </property>
          <property dictRef="cc:program">
            <scalar dataType="xsd:string">psi</scalar>
          </property>
          <property dictRef="cc:hostName">
            <scalar dataType="xsd:string">mac10147</scalar>
          </property>
          <property dictRef="cc:programVersion">
            <scalar dataType="xsd:string">4.0</scalar>
          </property>
        </propertyList>
      </module>
    </module>
  </module>
</module>
~
~
~
```

Currently works with CML Validator that checks schema, convention rules, dictionary references.

Next steps

1. Settle definitions of ‘jobList’, ‘job’ and ‘calculation’.
2. Continue adding psi4 properties to output; along with psi4 dictionary entries.
3. Collaborate with Avogadro 2 developer.
4. [Possible issue #3: Integrate properties and dictionary with underlying psi4 property code? Maybe someday?]