Some practical HINTS on how to use MOLCAS in quantum chemistry problems

Luis Serrano-Andrés and Valera Veryazov

Outline

- Some comments about different programs
- Planning your research
- Final admonitions and request

GATEWAY/SEWARD program

- Try the Choleski approximation (or RI)!
  It saves disk space and possibly time.
- Think about basis set. ANO-like basis sets have many advantages, but they are "marginal".
- Try to avoid inline basis sets, use the library.
- Remember that the quality of basis set should match the quality of computational method.
- Use ANO-RCC even for atoms in the 2nd row.
- Be extremely careful when computing anions.
  Remember that special situations require special basis sets.
- Use minimal or small basis set for understanding chemical problem.

SCF program

- HF orbitals are in many cases good starting orbitals, but quite often GuessOrb can be used instead.
- Very large basis set together with HF can lead to linear dependences.
- Remember! Hartree-Fock method allows multiple solutions (even for trivial systems)
- Be reasonable selecting convergence thresholds
- UHF convergence is poor comparing to RHF.
- Sometime you have to use Scramble keyword to break the symmetry.
- DFT convergence can't be better that HF convergence.
- Think about starting orbitals for DFT.

RASSCF program

- MCSCF are multi-solution methods that heavily depend on the starting orbitals
- On convergence ALWAYS (ALWAYS, ALWAYS, etc) check the orbitals!
  If your active space has been changed - your calculation might be meaningless
- Do not try to make calculation at once!
- Increase LEVShift parameter in cases of slow or difficult convergence.
- Use reasonable thresholds, and observe the convergence

RASSCF: roots

- RasOrb, rasscf.molden files contains average orbitals, numbered files contain natural orbitals for different roots.
- Try increasing the number of SA-CASSCF roots to locate more excited states.
- Do NOT play games with weights for the different roots. Roots with equal weights make your calculation more clear and reproducible.

RASSCF: symmetry

- Use symmetry with care! Especially if real symmetry did not match the symmetry in your calculation
- Start with clean orbitals. SCF might corrupt symmetry.
- Use, if needed, CLEANUP and SUPSYM, for linear molecules: LINEAR
- Working in a too high symmetry might prevent you from obtaining less symmetric lowest-lying localized solutions
- Try C1 in suspicious cases

RASSCF: RAS

- use it! RASSCF is a simple way to increase the active space.
- Balance RAS1/3 and RAS2 subspaces. Try to change orbitals between these subspaces
- Removing RAS2 space completely is not a good idea.
- Note that RAS calculations have a slower convergence, and demand more resources.

CASPT2

- PT2 is a correction. If it is large, you might be doing something wrong.
- Do not use parameters in CASPT2 (IPEA, levelshift) to reach an agreement with experiment!
- The new IPEA = 0.25 zeroth Hamiltonian is the default from Molcas-6.4. Note that when comparing with old results
- Do not compare calculations with different active space, or reference
- An intruder state (low reference weight in the CASPT2 state) might be informing you that your active space lacks an important orbital.
- For weakly interacting intruder states cases try the IMAGINARY level shift parameter. Beware of using too large level shifts (not larger than IMAGINARY 0.20)

multistate CASPT2

- For heavy valence-Rydberg mixing cases or for closely degenerated CASPT2 states, use MS-CASPT2
- If the MS-CASPT2 description differs a lot from the CASPT2 one, try to check the calculation by increasing the active space.
- The "true" solution is typically between both cases (CASPT2 and MS-CASPT2). If you are suspicious about the MS-CASPT2 result, better keep the CASPT2 one.

Active space selection

- Use your chemical intuition
- Always compare calculations with the same active space size
- Keep the active space during geometry change!
- Consult Björn Roos' rules for particular atoms
- Look at orbitals!
- use expbas! start from minimal basis set, decide the active space, and expand the basis to "normal". With small basis set you can clearly identify orbitals.
- use localization! Especially for virtual orbitals
- use RAS1/RAS3 spaces
- Experiment with your active space.

RASSI program

- For spin-orbit calculations, you may need more input states. Remember: the SO is obtained by sum-over-states method.
- Properties computed by default are approximate. Even a full calculation using DENS keyword is approximate, if based on JOBMIX input files.
- With JOBMIX input files use always EJOB keyword. Else, the Hamiltonian will be recomputed and the result will just reproduce the CASSCF.
- You can use many JOBIPH files from CASSCF, but only one for each state and symmetry of JOBMIX files from CASPT2. Do not use both JOBMIX and JOBIPH files in the same calculation.
- For advanced uses (e.g. constructing diabatic states) make sure you have understood the working of RASSI correctly.

Geometry optimization: planning

- plan your geometry optimization. it will take time!
- Not all codes in MOLCAS works with analytical gradients
- CASSCF does not include dynamic correlation, use DFT (or CASPT2) instead
- Try to change optimization problem into grid problem
- Check the convergence (aka Save the planet)
- Think about thresholds for your problem
- think about a good starting geometry

Geometry optimization: convergence

- for flat hypersurfaces decrease CUTOFF in ALASKA
- Poor active spaces may lead you to symmetry broken wrong solutions
- on poor convergence use keyword CARTESIAN or/and reduce trust radius (MAXSTEP)
- be very careful if your system is close to 'extra' symmetry
- to break symmetry - distort your system, or apply an electric field

Geometry optimization

- Linearly interpolated internal coordinates geometries may be a good starting point to locate a transition state.
- CASSCF surface crossing might be different to CASPT2.
- Numerical hessians and optimizations may lead you to bad solutions when different electronic states are too close.
- Be careful with the change of roots and nature along a geometry optimization

Hardware/software issues

How NOT to misuse MOLCAS?

- Do not play too much with code optimization.. we did it for you!
- use optimized BLAS libraries (e.g. MKL)
- remember: Memory can be more important than CPU speed.
- do not overload MOLCASMEM
- Solid state devices are very good for conventional integral codes, but they are average for CD/RI.

Parallel issues

- test GA before compiling molcas in parallel
- be very careful if run MOLCAS in parallel!
  Think: where do you run it, what memory is used, etc.
- MOLCAS has cross node parallelization.
  Using multicores can dramatically decrease the performance!
- think about MOLCASMEM (seriously!)
- GPUs can be used, but do not expect too much.
  They are designed for different purpose
- read about Amdahl's law to avoid disappointments

Final admonitions

Not matching experiments may sometimes be good news, either because we are comparing different things, or because our models do not consider all the required effect. We have to use our calculations to EXPLAIN and PREDICT the experimental data, not to simply have "good agreement".

Do not feel that theory is always inferior to experiment.

In many cases experimentalists need us more than we need them, because they don't know at all what they have measured.

Don't think that there is (somewhere) a superior method, which is always correct. It is not a method which makes results better or worse, it is we who use these methods in a right or wrong way.

The conclusion you made after calculation is not 'a number'.

Try to find a chemical explanation of the problem you study.

Final request

Please, help us to improve MOLCAS:

- Send us bug reports and suggestions through the web page
- Participate in the discussion forum: add questions and answers
- If you make useful tools related to MOLCAS for your personal use think of sharing them. They might be added to the package.
- Remember the developers that there is pending tools deeply needed. The list of priorities may change in that way.