Easy way to Conical Intersections

Alexander A. Granovsky

Laboratory of Chemical Cybernetics

M.V. Lomonosov Moscow State University, Moscow, Russia

November 12th , 2008

Caution

This document contains new information, know-how, trade secrets, etc… (*INFORMATION*) which is the copyrighted work of authorship owned by *The PC GAMESS/Firefly Project Team*. The copyright and other intellectual property rights in the *INFORMATION* are the exclusive property of *the PC GAMESS/Firefly Project Team. No parts of this INFORMATION* can be used without getting prior written permission from *the PC GAMESS/Firefly Project Team*

Approaches to Conical Intersections (CIs) studies

- State-averaged (SA) -MCSCF is the most widely used approach as of yet
  - It's the only approach considered in this presentation

What we actually need:

- CI location
  - $E_i=E_j$, $E_i=min$
  - One needs at least state-specific (SS) gradients for states of interest within the SA-MCSCF framework
  - If searching for CIs using non-adiabatic basis, one needs gradient of non-adiabatic coupling constant as well. However, one can avoid this by applying adiabatic basis
- Excited states optimization
  - $E_i=min$
  - One needs SS gradients for states of interest within the SA-MCSCF framework
- Our goal
  - **SS gradients for SA-MCSCF**

Basics of molecular gradients theory

- E = *E(C,**c**,**I**)*, where
  - *C* is the matrix of MO coefficients
  - ***c*** are the coefficient of “wavefunction” expansion over CSFs or determinants
  - ***I*** are 1-e and 2-e integrals in AO basis
- $dE/dx = \partial E/\partial C \cdot \partial C/\partial x + \partial E/\partial \mathbf{c} \cdot \partial \mathbf{c}/\partial x + \partial E/\partial I \cdot \partial I/\partial x$
  - basic equation of gradient theory

SS-MCSCF case

- $dE/dx = \partial E/\partial C \cdot \partial C/\partial x + \partial E/\partial \mathbf{c} \cdot \partial \mathbf{c}/\partial x + \partial E/\partial I \cdot \partial I/\partial x$
  - $\partial E/\partial C = 0$
  - $\partial E/\partial \mathbf{c} = 0$

- Thus: $dE/dx \equiv \partial E/\partial I \cdot \partial I/\partial x$
  - Just like for HF itself
- **The theory of molecular gradients is very simple for SS-MCSCF!**
  - **the implementation is very straightforward and efficient as well**

SA-MCSCF case

- $dE_i / dx = \partial E_i / \partial C \cdot \partial C / \partial x + \partial E_i / \partial \mathbf{c}_i \cdot \partial \mathbf{c}_i / \partial x + \partial E_i / \partial I \cdot \partial I / \partial x$
  - $\partial E / \partial C = 0$ *but* $\partial E_i / \partial C \neq 0$!
  - $\partial E_i / \partial \mathbf{c}_i = 0$

- Thus: $dE_i / dx = \partial E_i / \partial C \cdot \partial C / \partial x + \partial E_i / \partial I \cdot \partial I / \partial x$
  - Much more difficult case to handle
  - **Formally we need to solve MC-CPHF equations for z-vector to find MOs response terms**
    - **Need much wider class of transformed 2-e integrals**
      - **less efficient and slower integral transformation with possible integral sorting step**
      - **perhaps much longer step than MCSCF itself, esp. for large problems**
    - **Need to set up and solve large system of linear equations**
    - **Need to handle MCSCF of different types separately (e.g., CASSCF would be the special case)**

I’m too lazy actually…

- Are there any alternative ways to go?
  - Indeed they are…

The approach

- SA-MCSCF:
  - $E = \sum_i w_i E_i = E(\bar{w}, C, \bar{\mathbf{c}}, I)$
  - $\partial E / \partial w_i = E_i$
  - $dE / dw_i = \partial E / \partial w_i + \partial E / \partial C \cdot \partial C / \partial w_i + \partial E / \partial \mathbf{c} \cdot \partial \mathbf{c} / \partial w_i + \partial E / \partial I \cdot \partial I / \partial w_i$
    - where: $\partial E / \partial C \equiv 0, \partial E / \partial \bar{\mathbf{c}} \equiv 0, \text{and } \partial I / \partial w_i \equiv 0$
- **Finally:** $dE / dw_i = E_i$

The final step

- Considering the following identity:

$$dE_i / dx = d(dE / dw_i) / dx = d^2E / dw_i dx = d(dE / dx) / dw_i$$

- **we finally realize:**

$$\bar{g}_i = dE_i / dx = d\bar{g} / dw_i$$

- **where $\bar{g}$ is gradient of the SA-MCSCF energy**

How can we use this?

- The expression for $\bar{g}$ is very simple – exactly the same as for SS-MCSCF case above
  - **Any existing code capable to compute SS-MCSCF gradients can be used to compute gradient of SA-MCSCF energy as well**
- The expression for weight derivatives is not simple at all
  - Actually we just do not need it
    - **Let's differentiate with respect to weight numerically!**
    - **Practical approach – use of three-point finite difference formulas**
    - **Straightforward extension to any derivatives of any order, e.g., Hessians, response-type multipole moments, polarizabilities, etc…**

So what we got is:

- Traditional approach
  - Solve SA-MCSCF problem
  - Calculate SS gradient
    - Integral transformation
    - Solve MC-CPHF equations
    - And finally, gather all contributions to SS gradient, including AO part
- **Our approach**
  - **Solve two or three SA-MCSCF problems**
  - **Calculate SA gradient two (central differencing formulas) or three (non-symmetric formulas) times**
  - **The rest is just a simple math**

Is this affordable?

- Computational demands
  - **Our way is the fastest possible for large AO basis sets**
    - **Does not require large-scale four-center integrals transformation**
    - **Does not require solution of large system of linear equations**
- Precision
  - **Resulting approximate gradients are smooth functions of (geometric) parameters!**
  - **Numerous numerical experiments show that use of Δw of about 10^{-3} results in gradients of six-digits accuracy**
    - **Numerically stable**
    - **Enough for geometry optimization and CIs location**
    - **Enough for semi-numerical Hessians**
    - **Seems to be enough even for double-seminumerical Raman activities!**

It’s a little bit tricky however…

- No known programs are capable to deal with SA-MCSCF energies and gradients in the case of non-unit sum of weights
  - **Simple solution to avoid code changes:**
    - **Perturb the weights**
    - **Normalize them back**
    - **Perform calculation**
    - **Renormalize the answer**
- Extra high precision is needed while solving SA-MCSCF problem
  - **It’s easy, fast, and efficient with PC GAMESS/Firefly**
  - **For standard way, we would need some extra precision as well**

It's a little bit tricky however…

- Excited states geometry optimization
  - No (quasi) degeneracy in most cases
  - **Central (symmetric) finite difference formula is the best**
    - **Is numerically stable**
    - **Requires only two SA gradient calculations per SS gradient**
    - **Somewhat higher precision**
- CIs location
  - Near or at (quasi) degeneracy
  - **Central finite difference formula is not applicable anymore**
    - **Is not numerically stable because SA-MCSCF may have multiple solutions (branches) for slightly differently "weighted" calcs.**
    - **Use of one-sided finite differencing formulas solves the problem!**
      - **Minor overhead – third SA gradient calculation is required**

Concluding remark

- CI location
  - Most common situation – two-state averaged MCSCF
    - **We do not need six MCSCF computations to find two SS gradients**
    - **All the required information can be obtained while computing SS gradients for any single state**

Thank you for your attention!