

Estimating Cyclic Peptide Ring Strain in Python with PySCF

Let $\mathbf{R} \in \mathbb{R}^{3N}$ denote the nuclear coordinates of a peptide and let $E(\mathbf{R})$ denote the Born–Oppenheimer ground–state electronic energy obtained from an electronic structure method (HF/DFT/MP2/CCSD, etc.). We distinguish between two manifolds of geometries:

$$\mathcal{M}_{\text{ref}} = \{\mathbf{R} : \text{open/capped reference peptide}\}, \quad (1)$$

$$\mathcal{M}_{\text{cyc}} = \{\mathbf{R} : \text{head-to-tail cyclic peptide}\}. \quad (2)$$

Geometry optimization at a given level of theory yields

$$E_{\text{ref}} = \min_{\mathbf{R} \in \mathcal{M}_{\text{ref}}} E(\mathbf{R}), \quad (3)$$

$$E_{\text{cyc}} = \min_{\mathbf{R} \in \mathcal{M}_{\text{cyc}}} E(\mathbf{R}). \quad (4)$$

The *ring strain* is defined as the energetic penalty of enforcing the cyclic topology:

$$\boxed{\Delta E_{\text{strain}} = E_{\text{cyc}} - E_{\text{ref}}}. \quad (5)$$

In the harmonic approximation, if $\mathbf{R}_{\text{ref}}^*$ and $\mathbf{R}_{\text{cyc}}^*$ denote the optimized reference and cyclic geometries and $\delta\mathbf{R} = \mathbf{R}_{\text{cyc}}^* - \mathbf{R}_{\text{ref}}^*$, then near $\mathbf{R}_{\text{ref}}^*$,

$$\Delta E_{\text{strain}} \approx \frac{1}{2} \delta\mathbf{R}^T H(\mathbf{R}_{\text{ref}}^*) \delta\mathbf{R}, \quad (6)$$

where H is the Hessian matrix. Thus ring strain corresponds to the elastic energy stored in the distortion of bond lengths, angles, and torsions necessary to achieve ring closure and accommodate steric interactions.

Computational Procedure

For each system (cyclic and reference forms), we perform the following workflow:

1. **3D structure generation.** Initial coordinates are generated from SMILES using RDKit and pre-optimized with MMFF or UFF to obtain a chemically reasonable starting point.
2. **DFT geometry optimization.** Each structure is optimized at the DFT level (e.g. WB97X/def2-SVP) using PySCF’s `geometric_solver`. This yields optimized geometries $\mathbf{R}_{\text{cyc}}^*$ and $\mathbf{R}_{\text{ref}}^*$.
3. **Single-point electronic energies.** On the DFT-optimized structures, we compute

$$E_{\text{cyc}} = E(\mathbf{R}_{\text{cyc}}^*), \quad E_{\text{ref}} = E(\mathbf{R}_{\text{ref}}^*),$$

4. **Ring strain per method.** For each method M , we evaluate

$$\Delta E_{\text{strain}} = E_{\text{cyc}} - E_{\text{ref}}.$$

Density Functional Theory (DFT) 101

Density functional theory (DFT) has been massively successful for computational chemistry. DFT is able to tackle up to million atom simulations. DFT is capable of achieving such remarkable feats due to how it finds a short cut to computing observables, most notably the total energy (ground state energy) which is essential for understanding the chemistry of the system. This short cut is instead of trying to solve the η -particle Schrödinger equation

$$H\psi_0(r_1, \dots, r_\eta) = E_0\psi_0(r_1, \dots, r_\eta), \quad (7)$$

we instead try to find an *electron density* $n(r)$ that reproduces the values for $E_0(r)$ we would obtain if we did solve the full η -particle system where

$$n(r) = \eta \int dr_2 \cdots dr_\eta \|\psi(r, r_2, \dots, r_\eta)\|^2 \quad (8)$$

which corresponds to the density of the η electrons in space as determined by the underlying wave function. In order to solve for the density $n(r)$, we must iteratively solve the 1-particle Kohn-sham equations

$$\left[-\frac{\nabla^2}{2} + u_s[n](r) \right] \phi_i(r) = \epsilon_i \phi_i(r) \quad (9)$$

and compute the density

$$n(r) = \sum_{i=1}^{\eta} \|\phi_i(r)\|^2. \quad (10)$$

When solving this system of non-interacting particles, we need the effective potential function $u_s[n](r)$ which is a functional of the density $n(r)$ and can be written as

$$u_s[n](r) = u_0[n](r) + u_H[n](r) + u_{xc}[n](r) \quad (11)$$

where u_0 is the external potential, u_H is the classical Coloumb energy, and u_{xc} is the exchange–correlation potential. The success or failure of the DFT algorithm is ultimately determined by the choice of exchange functional that is used to approximate

$$u_{xc}[n](r) = \frac{\delta E_{\text{ex}}[n]}{\delta n(r)}, \quad (12)$$

where $E_{\text{ex}}[n]$ is the exchange–correlation energy functional. There is a huge library of functionals that can be used to implement $E_{\text{ex}}[n]$ in the DFT algorithm. Choosing the right one is a non-trivial task and if the chemical system of interest exhibits strong correlation, the DFT trick breaks down entirely and gives inaccurate energies. Once a functional is selected, an initial guess at a density $n_0(r)$ is made and the Kohn-Sham equations are iteratively solved, ultimately converging the target particle density $n(r)$.

The important takeaway is that the effectiveness of DFT is dependent on the choice of exchange–correlation functional. Perdew’s “Jacob’s ladder” is a conceptual hierarchy of exchange–correlation approximations in Kohn-Sham DFT

[?]. Each “rung” adds more information about the electronic system, aiming to climb from simple, local descriptions of the electron gas toward the “heaven” of the exact functional (i.e. chemical accuracy for real systems). The ladder also reflects a practical trade-off: higher rungs are typically more accurate but also more expensive and sometimes less robust (e.g. for metals).

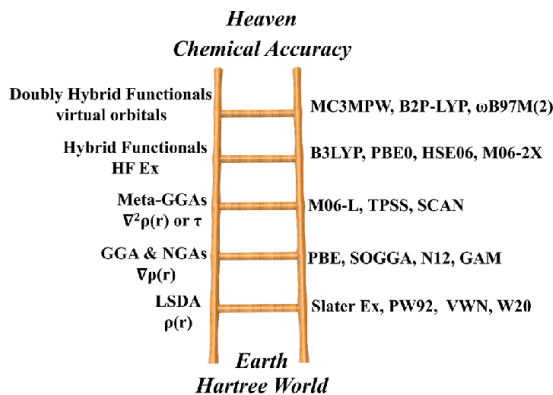


Figure 1: Jacob’s Ladder of DFT exchange-correlation functionals

Rung 1: Local Density Approximation (LDA)

- Uses only the electron density $\rho(\mathbf{r})$.
- Assumes the system is locally like a uniform electron gas.
- Works surprisingly well for bulk metals and slowly varying densities.

Rung 2: Generalized Gradient Approximation (GGA)

- Uses $\rho(\mathbf{r})$ and its gradient $\nabla\rho(\mathbf{r})$.
- Can distinguish inhomogeneous regions and improves binding and molecular energetics over LDA.
- Typical workhorse functionals in surface science and chemistry.

Rung 3: meta-GGA

- Uses ρ , $\nabla\rho$, and an additional orbital-dependent ingredient such as the kinetic energy density $\tau(r) = \frac{1}{2} \sum_i |\nabla\psi_i(r)|^2$ or the Laplacian $\nabla^2\rho$.
- Better ability to discriminate different bonding situations and densities.

Rung 4: Hybrid Functionals

- Add a fraction of exact (Hartree–Fock) exchange on top of semilocal ingredients.

- Reduce self-interaction and delocalization errors; often improve reaction barriers and band gaps.
- **Such as WB97X used for computing ring strains.**

Rung 5: Double Hybrids

- Combine hybrid DFT with perturbative correlation (often MP2-like) built from Kohn–Sham orbitals.
- Highest accuracy in the ladder, but also highest computational cost.