Learning exchange-correlation functional with **differentiable density functional theory**

Muhammad F. Kasim
Sam M. Vinko
muhammad.kasim@physics.ox.ac.uk
sam.vinko@physics.ox.ac.uk

https://arxiv.org/abs/2102.04229

We wrote a 3D fully differentiable density functional theory (DFT) simulation and use it to improve simulation accuracy

Motivations

- Accurate simulations of molecular/material properties are keys in advancing novel material discovery.
- One of the most widely used quantum chemistry simulations, DFT, is accurate enough for some cases, but insufficient for some modern use cases.
- Can we use machine learning to improve the accuracy of DFT using **limited experimental data**?

Observations

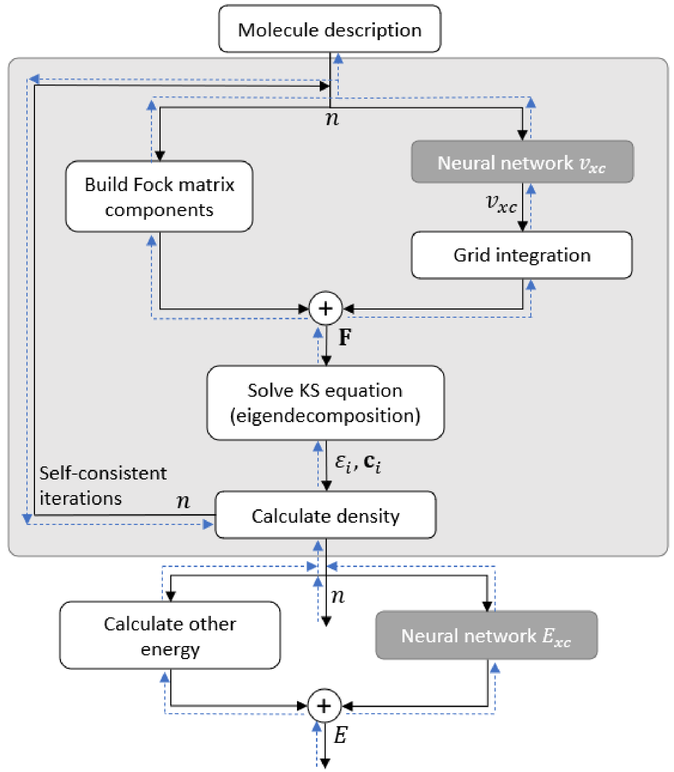
- DFT accuracy depends largely on the exchange-correlation (xc) functional.
- Xc functional takes electron density, $n(\mathbf{r})$, as its input, and the energy as its output, i.e. $E_{xc}[n(\mathbf{r})]$.
- Improving xc functional also improves the DFT accuracy.

Problems

- There is no direct experimental data on xc functionals (only on molecular/material properties).
- Experimental data on molecular properties are **limited and heterogeneous**.

Ideas

- Use a neural network to represent the xc functional (XCNN).
- Wrap the XCNN with a **fully-differentiable DFT** to enable learning from molecular properties.



Schematics of the differentiable DFT

Basic DFT procedures

- Construct the Hamiltonian matrix components by evaluating the **Gaussian integrals using libcint** library (Sun, 2014).
- Perform **eigen-decomposition** on the Hamiltonian matrix.
- Repeat the procedures until **self-consistency** (or equilibrium) achieved.

How it's made differentiable

- **Wrap libcint with PyTorch**. The gradient expressions are easily derived.
- For non-degenerate case, it is available on PyTorch. For degenerate case, we **follow Kasim (2020) for numerical stability**.
- Use **implicit theorem** to calculate the gradient.

Training data

- Has only 12 experimental data + 15 simulated data for regularization.
- Consists of molecules with **1 – 2 atoms** only.
- Only contains **H – Ne atoms**.

Type	Atomization energy	Density profile	Ionization potential
Training	H_2, LiH, O_2, CO	H, Li, Ne, H_2, Li_2, LiH, B_2, O_2, CO	O, Ne
Validation	N_2, NO, F_2, HF	He, Be, N, N_2, F_2, HF	N, F

Test data

- Ionization potential (IP) on 18 atoms (H – Ar).
- Atomization energy on 104 molecules:
  - HC: Hydrocarbons (up to **14 atoms** in a molecule)
  - SHC: Substituted hydrocarbons
  - NHC-1: Non-hydrocarbons with only H – Ne atoms
  - NHC-2: Non-hydrocarbons containing Na – Ar atoms

Error on molecular properties predictions

Calculation	IP	AE	AE HC	AE SHC	AE NHC-1	AE NHC-2
Local Density Approximations (LDA)						
LDA (Perdew & Wang, 1992)	**6.9**	70.4	97.0	101.1	58.8	43.7
XCNN-LDA	50.8	**15.3**	**22.7**	**19.4**	**7.8**	**16.6**
XCNN-LDA-IP	15.2	18.5	25.4	21.8	8.4	22.9
Generalized gradient approximations (GGA)						
PBE (Perdew et al., 1996)	**3.6**	16.5	15.1	23.2	18.0	9.8
XCNN-PBE	10.7	**7.4**	**5.4**	**8.4**	**6.5**	**8.5**
XCNN-PBE-IP	4.1	8.1	6.8	9.6	6.7	8.6
Other approximations						
SCAN (Sun et al., 2015)	3.7	5.2	3.8	8.2	4.0	4.6
CCSD (basis: cc-pvqz)	2.0	12.1	11.1	17.7	11.2	9.0
CCSD-T (basis: cc-pvqz)	**1.3**	**3.5**	**2.2**	**5.6**	**2.5**	**3.5**

Conclusions

- XCNN improves the predictions on **larger molecules** and **molecules with atoms not present in the training**.
- This paves a new way to improve DFT accuracy with machine learning

References
Qiming Sun (2014), arxiv: 1412.0649
Muhammad F. Kasim (2020), arxiv: 2011.04366