A PYTHON PACKAGE FOR ORGANIC INTERFACES GENERATING

--By Jiawei Zhan

A BRIEF INTRODUCTION TO THE PROJECT

- The project was mainly about efficiently predicting the structure of organic and inorganic interfaces, which could be divided into three parts.

1. cleaving slabs
2. lattice matching (my part)
3. surface matching

SOME HIGHLIGHTS ON MY WORK

Rough orientation-searching algorithm Implemented before coming to CMU:

- Questions or goals:
- Used pure geometric method to shorten the time the program would spend on predicting the possible orientations of two different molecules in heteroepitaxy's growing.
- Methods and Algorithms:

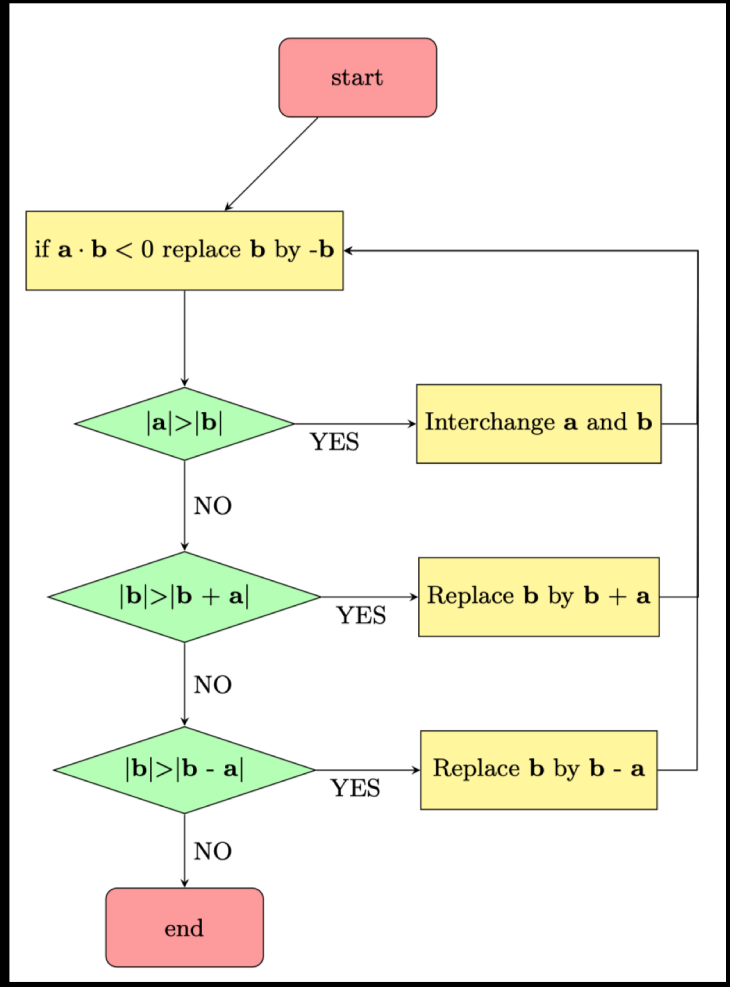
1. Present lattice in a common way
2. Match lattice in a common way

Rough orientation-searching algorithm Implemented before getting to CMU:

1.Present lattice in a common way

❖CLUES:

- Only intrinsic properties of lattices would be used to determine the superlattices without any reference to a particular coordination system, which means that any rotation or reflection would have no influence on the presentation.



Procedure to find a common expression of different lattice

Rough orientation-searching algorithm Implemented before coming to CMU:

2. Match lattice in a common way

❖ CLUES :

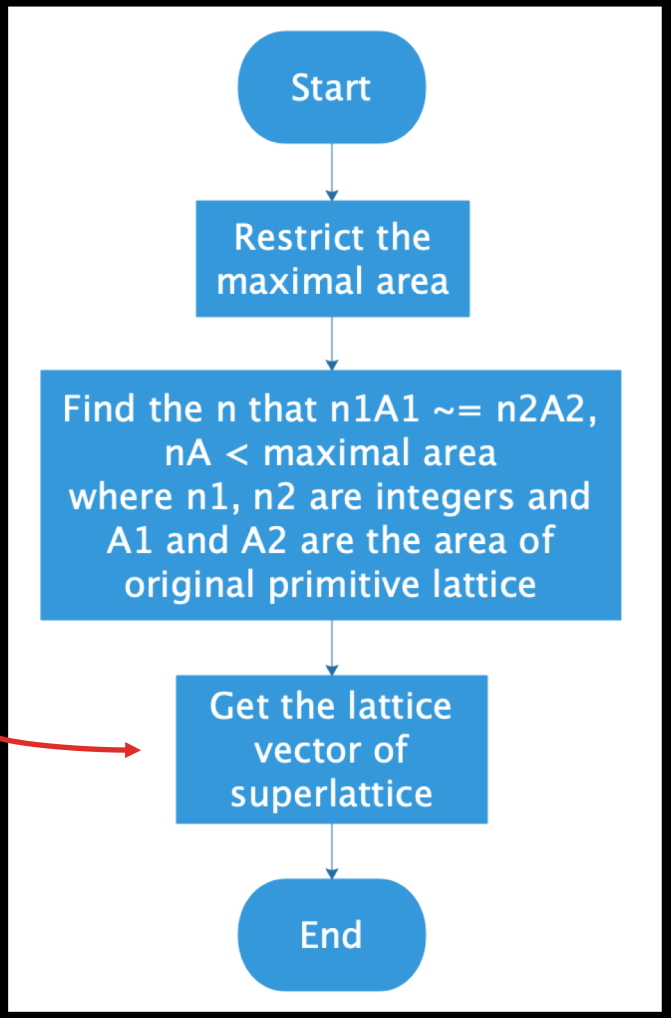
- Two lattice match only if their superlattice possess the same or almost the equal lattice vectors.

The procedure of finding the lattice vectors of superlattices

$$\begin{pmatrix}\mathbf{u}\\ \mathbf{v}\end{pmatrix} = \begin{pmatrix} i & j \\ 0 & m \end{pmatrix} \begin{pmatrix}\mathbf{a}\\ \mathbf{b}\end{pmatrix}$$

with i, j, m integers, and

$$\begin{cases} i \cdot m = n \\ i, m > 0 \\ 0 \le j \le m-1 \end{cases}$$



Rough orientation-searching algorithm Implemented before coming to CMU:

3. Result

- GaAs on CdTe

Matrhing faces	Epitaxial condition	Cell area	GaAs			CdTe					
GaAs/CdTe	GaAS || CdTe	(Å^2)	a(Å)	b(Å)	$\alpha\,(degree)$	a(Å)	b(Å)	$\alpha\,(degree)$	$\%_a$	$\%_b$	$\%_\alpha$
(100) face / (100) face	$[0\bar{1}5]\|\|[0\bar{1}2]$	207.0	14.41	14.41	90.00	14.50	14.50	90.00	0.6	0.6	0.0
(100) face / (100) face	$[0\bar{3}5]\|\|[0\bar{5}1]$	271.0	16.48	16.48	90.00	16.53	16.53	90.00	0.3	0.3	0.0
(100) face / (111) face[a]	$[0\bar{1}1]\|\|[1\bar{2}1]$[a]	127.0	8.00	16.48	75.96	7.94	16.53	76.10	0.7	0.3	0.2
(100) face / (111) face	$[0\bar{1}1]\|\|[2\bar{1}1]$	255.0	15.99	16.48	75.96	15.88	16.53	76.10	0.7	0.3	0.2
(100) face / (111) face	$[0\bar{1}1]\|\|[1\bar{2}1]$	255.0	8.00	31.98	90.00	7.94	32.09	90.00	0.7	0.3	0.0
(100) face / (111) face	$[0\bar{5}3]\|\|[1\bar{4}3]$	383.0	16.48	23.99	75.96	16.53	23.82	76.10	0.3	0.7	0.2
(100) face / (111) face	$[0\bar{1}1]\|\|[1\bar{2}1]$	383.0	8.00	48.14	85.24	7.94	48.30	85.28	0.7	0.3	0.1
(110) face / (100) face	$[\bar{3}32]\|\|[0\bar{1}5]$	316.0	16.48	19.59	78.58	16.53	19.45	78.69	0.3	0.7	0.1
(110) face / (100) face	$[1\bar{1}1]\|\|[0\bar{1}1]$	316.0	13.85	23.31	78.58	13.75	23.37	78.69	0.7	0.3	0.1
(110) face / (110) face	$[1\bar{1}0]\|\|[1\bar{1}1]$	90.0	8.00	11.31	90.00	7.94	11.23	90.00	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}0]\|\|[1\bar{1}1]$	180.0	8.00	22.62	90.00	7.94	22.46	90.00	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}1]\|\|[1\bar{1}2]$	180.0	13.85	13.85	70.53	13.75	13.75	70.53	0.7	0.7	0.0
(110) face / (110) face	$[001]\|\|[2\bar{2}1]$	180.0	11.31	15.99	90.00	11.23	15.88	90.00	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}0]\|\|[1\bar{1}1]$	271.0	8.00	33.92	90.00	7.94	33.69	90.00	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}1]\|\|[1\bar{1}2]$	271.0	13.85	19.59	90.00	13.75	19.45	90.00	0.7	0.7	0.0
(110) face / (110) face	$[001]\|\|[\bar{2}21]$	271.0	11.31	23.99	90.00	11.23	23.82	90.00	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}0]\|\|[1\bar{1}1]$	361.0	8.00	45.23	90.00	7.94	44.92	90.00	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}1]\|\|[1\bar{1}2]$	361.0	13.85	26.52	79.98	13.75	26.33	79.98	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}0]\|\|[\bar{1}11]$	361.0	15.99	23.99	70.53	15.88	23.82	70.53	0.7	0.7	0.0
(110) face / (110) face	$[001]\|\|[2\bar{2}1]$	361.0	11.31	31.98	90.00	11.23	31.76	90.00	0.7	0.7	0.0
(110) face / (110) face	$[1\bar{1}0]\|\|[1\bar{1}1]$	361.0	15.99	22.62	90.00	15.88	22.46	90.00	0.7	0.7	0.0
(110) face / (110) face	$[2\bar{2}1]\|\|[001]$	361.0	19.59	19.59	70.53	19.45	19.45	70.53	0.7	0.7	0.0
(111) face / (111) face	$[0\bar{1}1]\|\|[\bar{1}12]$	55.0	8.00	8.00	60.00	7.94	7.94	60.00	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{1}0]\|\|[1\bar{2}1]$	110.0	8.00	13.85	90.00	7.94	13.75	90.00	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{1}0]\|\|[1\bar{2}1]$	166.0	8.00	21.16	79.11	7.94	21.01	79.11	0.7	0.7	0.0
(111) face / (111) face	$[\bar{1}12]\|\|[0\bar{1}1]$	166.0	13.85	13.85	60.00	13.75	13.75	60.00	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{1}0]\|\|[1\bar{2}1]$	221.0	8.00	27.70	90.00	7.94	27.51	90.00	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{2}1]\|\|[1\bar{1}0]$	221.0	13.85	15.99	90.00	13.75	15.88	90.00	0.7	0.7	0.0
(111) face / (111) face	$[0\bar{1}1]\|\|[\bar{1}12]$	221.0	15.99	15.99	60.00	15.88	15.88	60.00	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{1}0]\|\|[1\bar{2}1]$	276.0	8.00	34.85	83.41	7.94	34.61	83.41	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{2}1]\|\|[\bar{1}01]$	276.0	13.85	21.16	70.89	13.75	21.01	70.89	0.7	0.7	0.0
(111) face / (111) face	$[4\bar{1}5]\|\|[0\bar{1}1]$	290.0	18.32	18.32	60.00	18.34	18.34	60.00	0.1	0.1	0.0
(111) face / (111) face	$[1\bar{1}0]\|\|[1\bar{2}1]$	332.0	8.00	41.55	90.00	7.94	41.26	90.00	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{2}1]\|\|[1\bar{1}0]$	332.0	13.85	23.99	90.00	13.75	23.82	90.00	0.7	0.7	0.0
(111) face / (111) face	$[\bar{1}01]\|\|[2\bar{1}1]$	332.0	15.99	21.16	79.11	15.88	21.01	79.11	0.7	0.7	0.0
(111) face / (111) face	$[0\bar{1}1]\|\|[\bar{2}35]$	346.0	19.99	19.99	60.00	19.98	19.98	60.00	0.0	0.0	0.0
(111) face / (111) face	$[1\bar{1}0]\|\|[1\bar{2}1]$	387.0	8.00	48.64	85.28	7.94	48.30	85.28	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{2}1]\|\|[1\bar{1}0]$	387.0	13.85	28.83	76.10	13.75	28.63	76.10	0.7	0.7	0.0
(111) face / (111) face	$[1\bar{3}2]\|\|[1\bar{5}4]$	387.0	21.16	21.16	60.00	21.01	21.01	60.00	0.7	0.7	0.0

TABLE I. Good lattice matches of GaAs on CdTe. The primitive common unit cells in this table do not exceed 400Å^2, and the mismatch is less than 1%. Under these conditions, all the possible matches ofCdTe(1OO), (110), and (111) on GaAs(1OO), (110) and (111) are given in this table. For each possible match, we give here the epitaxial condition, as well as the common unit cell dimensions on each side of the interface. The epitaxial condition is a pair of crystal directions, one on each side of the interface, that will be parallel to each other. Many such pairs are possible, and only one of them is given here. The cells' dimensions in angstroms and degrees are given here for comparison. The mismatch percentage in all the three dimensions of the common unit cell is given in the last three columns. By the way, [a] is observed experimentally by J. T. Cheung[3]

Rough orientation-searching algorithm Implemented before coming to CMU:

3. Result

- CdTe on sapphire

Matrhing faces	Epitaxial condition	Cell area	CdTe			Al_2O_3					
$CdTe/Al_2O_3$	CdTe || Al_2O_3	(Å^2)	a(Å)	b(Å)	$\alpha\,(degree)$	a(Å)	b(Å)	$\alpha\,(degree)$	$\%_a$	$\%_b$	$\%_\alpha$
(100) face / (111) face	$[0\bar{5}3]\|\|[1\bar{1}0]$	546.0	18.90	28.99	85.60	18.94	28.87	85.23	0.2	0.4	0.4
(100) face / (111) face	$[01\bar{5}]\|\|[\bar{1}\bar{1}2]$	546.0	16.53	33.06	90.00	16.44	33.15	89.93	0.5	0.3	0.1
(100) face / (111) face	$[0\bar{5}1]\|\|[2\bar{1}\bar{1}]$	588.0	16.53	35.80	83.88	16.41	35.83	83.50	0.7	0.1	0.5
(110) face / (101) face	$[\bar{2}21]\|\|[\bar{1}21]$	505.0	11.23	45.15	85.24	11.30	45.32	85.06	0.6	0.4	0.2
(110) face / $(0\bar{1}1)$ face	$[1\bar{1}0]\|\|[011]$	534.0	9.17	58.53	85.51	9.10	58.84	85.85	0.8	0.5	0.4
(110) face / $(0\bar{1}1)$ face	$[001]\|\|[111]$	534.0	12.97	41.26	90.00	13.02	41.02	90.00	0.4	0.6	0.0
(111) face / (111) face	$[1\bar{4}3]\|\|[2\bar{1}\bar{1}]$	236.0	16.53	16.53	60.00	16.41	16.42	60.11	0.7	0.6	0.2
(111) face / (111) face	$[1\bar{4}3]\|\|[1\bar{2}1]$	473.0	16.53	28.63	90.00	16.42	28.45	89.86	0.6	0.6	0.2
(111) face / (111) face	$[\bar{1}\bar{1}2]\|\|[1\bar{1}0]$	491.0	23.82	23.82	60.00	23.68	23.71	60.10	0.6	0.5	0.2
(211) face / (101) face	$[1\bar{1}\bar{1}]\|\|[\bar{1}21]$	411.0	11.23	36.67	90.00	11.30	36.55	89.89	0.6	0.3	0.1

TABLE II. Good lattice matches of CdTe on sapphire. The primitive common unit cells in this table do not exceed 600Å^2, and the mismatch is less than 1%. Under these conditions, all the possible matches of CdTe(100), (110), (111), (210), (211) or (221) on $Al_2O_3(101)$, $(0\bar{1}1)$ or (111) are given here. The sapphire faces and directions in this table refer to the rhomobohedral notation.

SOME HIGHLIGHTS ON MY WORK

Designed a graph-theory algorithm for efficient slab generation

- Questions or goals:
- How to conserve the atoms' numbers and mechanical properties during the slab generation?
- Methods and Algorithms:

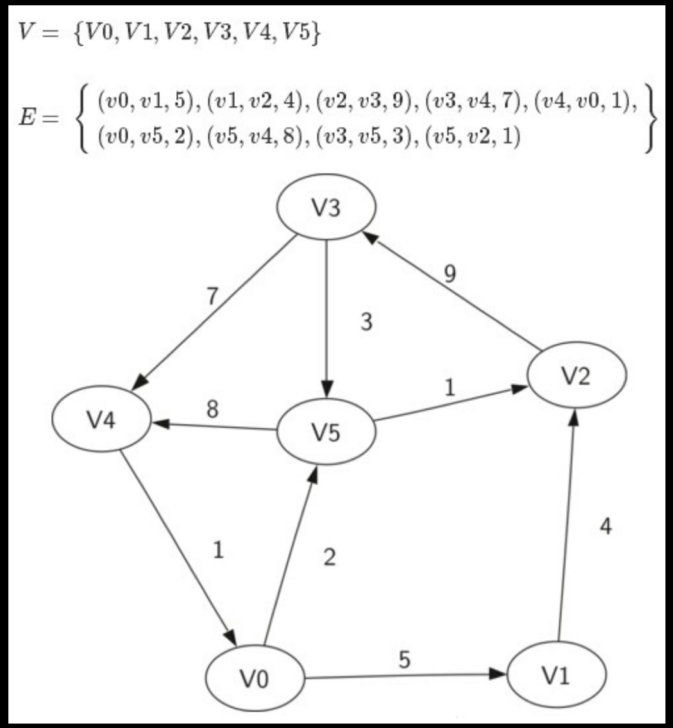
1. Designed a graph-theory algorithm, which regards molecules as subgraphs (nodes are atoms and edges are any relationships, and here the relationships are distances)
2. Find the connections between broken molecules and intact molecules, and try to selectively repair the broken molecules

Designed a graph-theory algorithm for efficient slab generation

1. Designed a graph-theory algorithm

❖ CLUES :

- One critical issue is that the cartesian coordinates of all atoms would change after cleaving slabs, which almost wraps all the useful information used in the methods that only based on the coordinates of atoms.
- In order to solve this problem, we need to transform the original Cartesian coordinates to a new coordinates that is invariant with respect to a rotation or translation of the system.
- Here, we use graphic structure to describe the molecules, which means that only the species of atoms and the distance between close atoms would be the parameters.



A directed connected graph

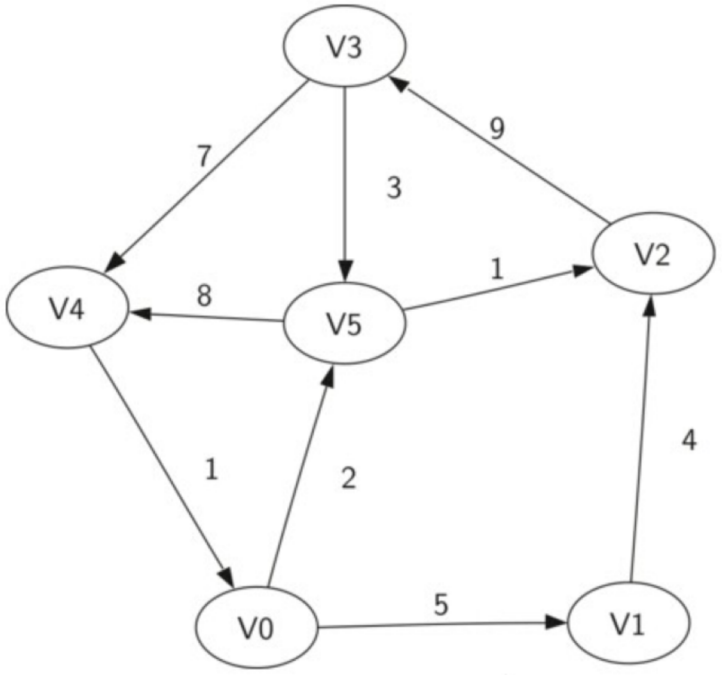
- Graph theory is the study of graphs, which are mathematical structures used to model pairwise relations between objects.
- In the graph theory, we could easily find out whether there is bijection in two graphs (graph isomorphism)
- We could shorten the time with Depth-First-Search algorithm and Breadth-First-Search algorithm.

Designed a graph-theory algorithm for efficient slab generation

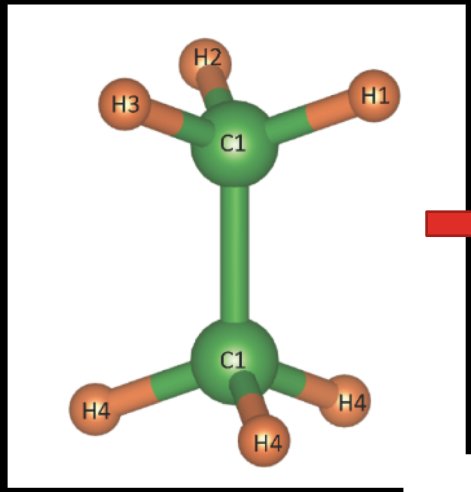
1. Designed a graph-theory algorithm

$$V = \{V0, V1, V2, V3, V4, V5\}$$

$$E = \left\{ \begin{array}{l} (v0, v1, 5), (v1, v2, 4), (v2, v3, 9), (v3, v4, 7), (v4, v0, 1), \\ (v0, v5, 2), (v5, v4, 8), (v3, v5, 3), (v5, v2, 1) \end{array} \right\}$$

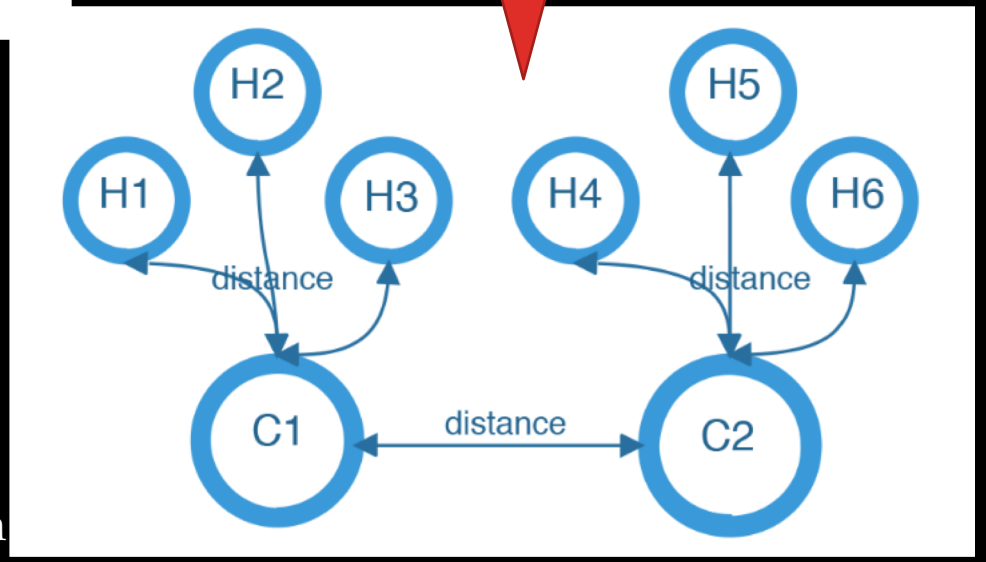


A directed connected graph



3D model

C2H6's transformation



2D connected graph

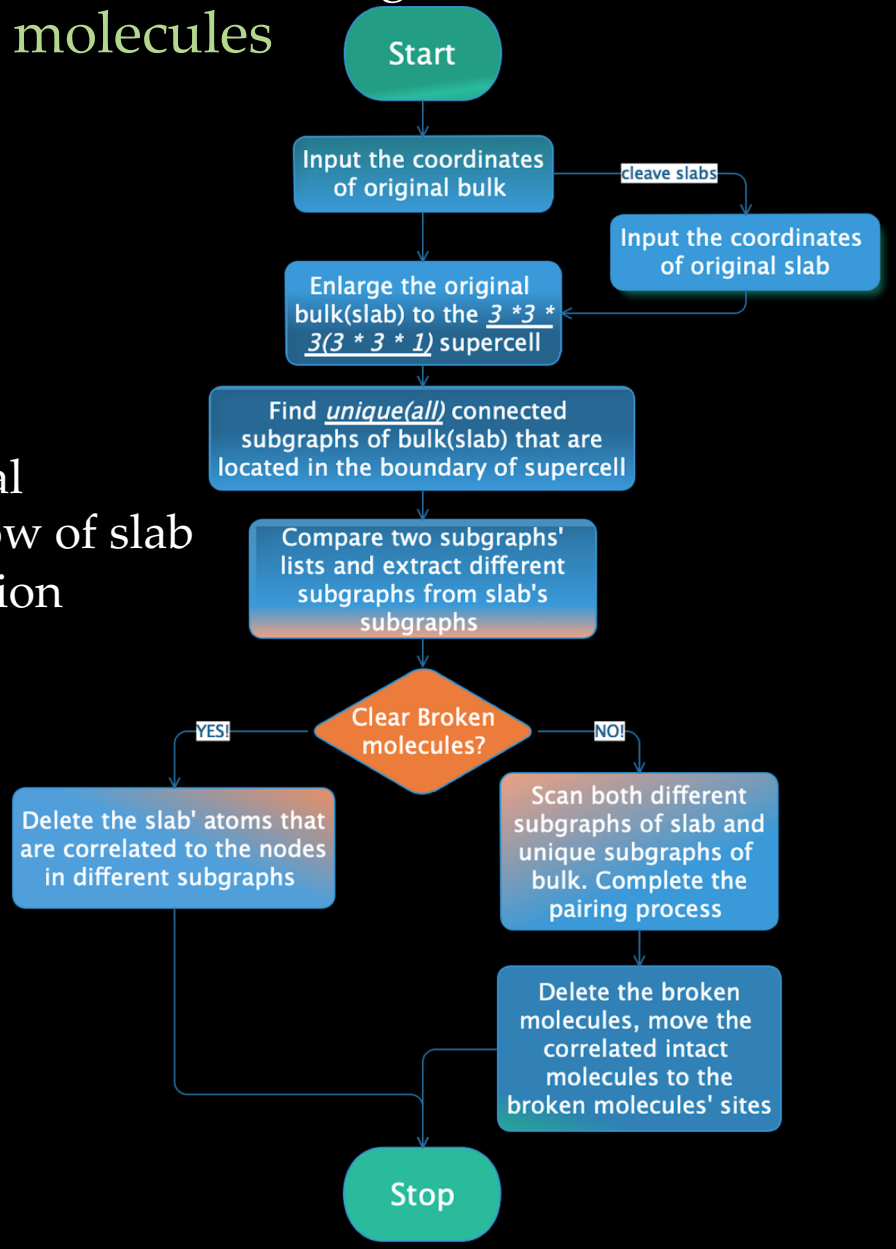
Designed a graph-theory algorithm for efficient slab generation

2. repair the broken molecules

❖CLUES :

- Find the connections between broken molecules and intact molecules, or whether there are inclusion relationship between the two subgraphs.
- To be more specific, we need to find the graph-pairs that the nodes of one are completely included by those of another, and the distances from the corresponding nodes to their corresponding nearest neighbors are the same

The total workflow of slab generation

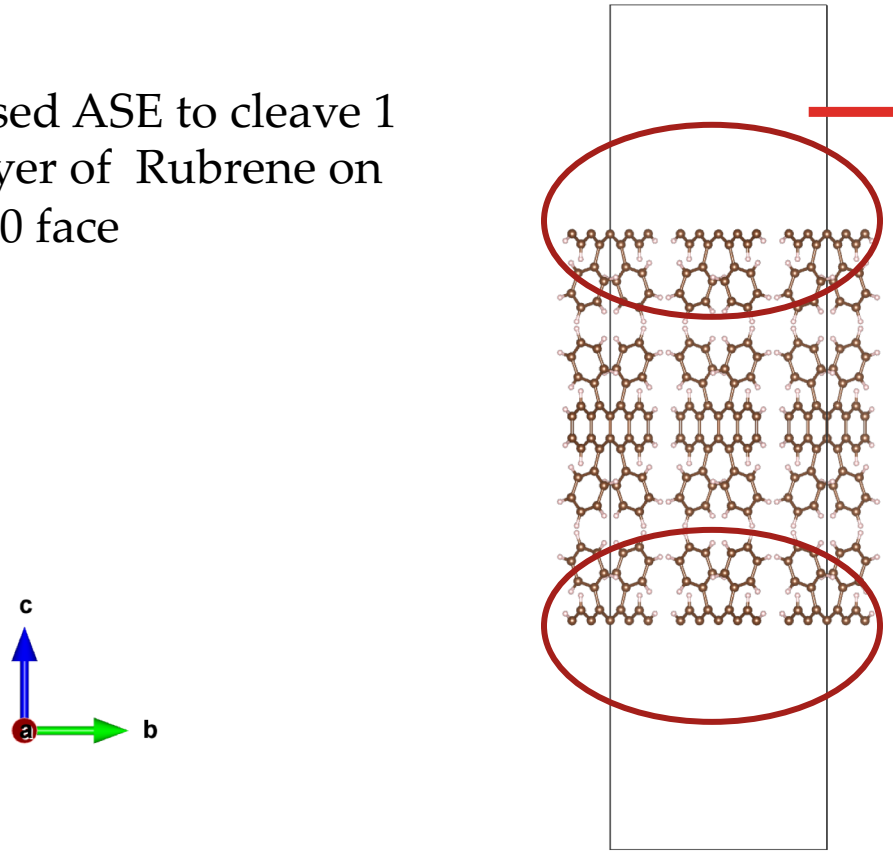


Designed a graph-theory algorithm for efficient slab generation

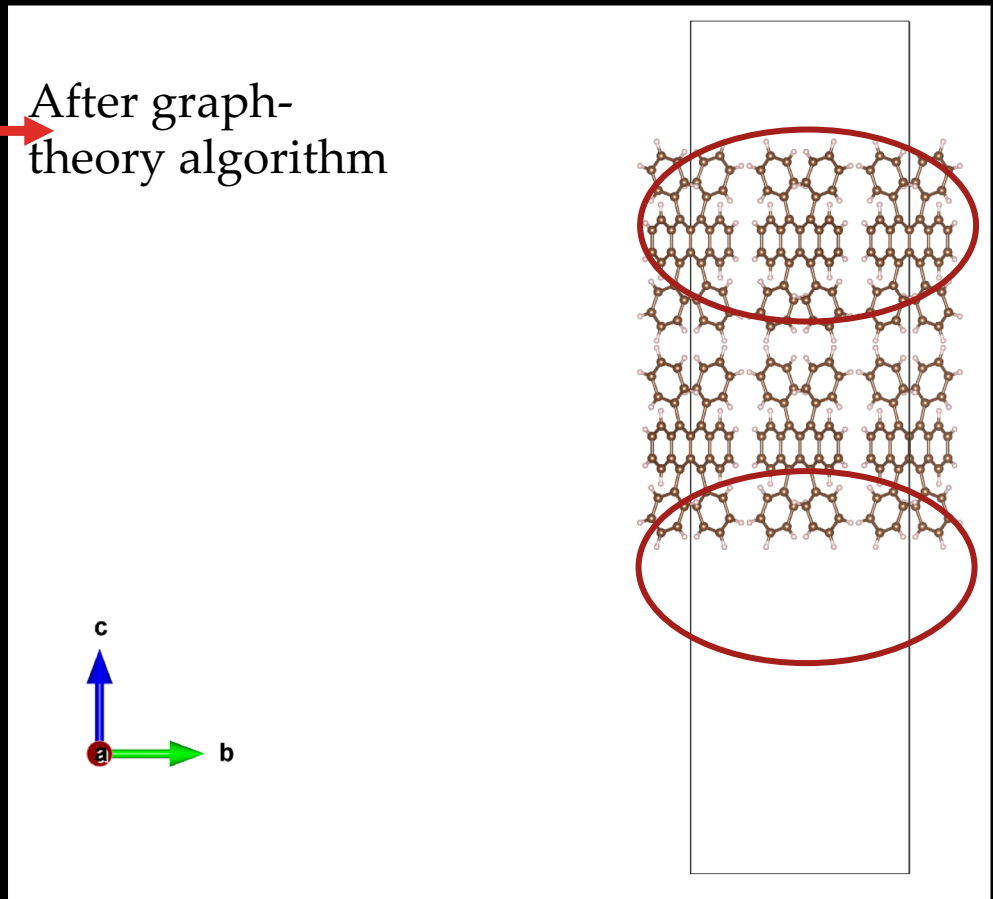
3. Result

- outperforming all previous algorithm in conserving the molecular number and structures as well as mechanical properties

Used ASE to cleave 1 layer of Rubrene on 100 face



After graph-theory algorithm



SOME HIGHLIGHTS ON MY WORK

Used empirical Forced Field to predict the structure of new interfaces in 3D space

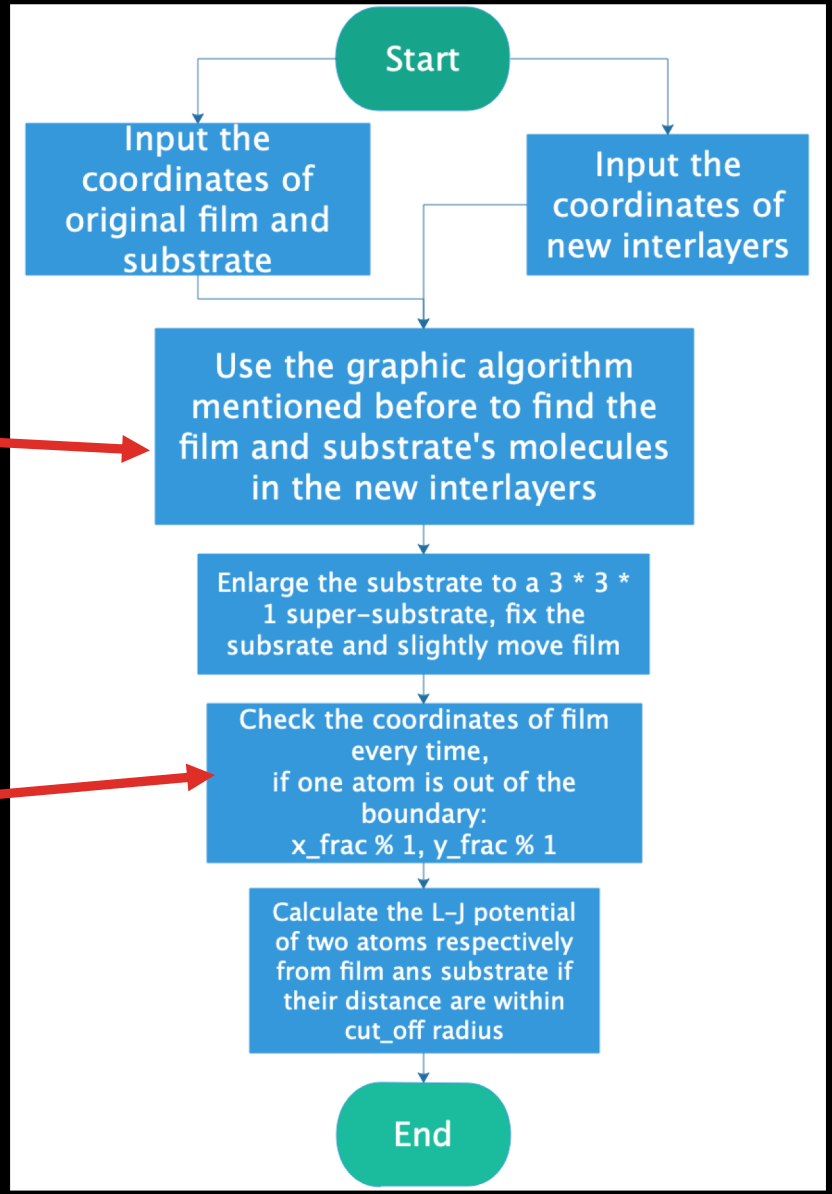
- Questions or goals:
- How to predict the energy of interfaces as precisely as possible?
- Methods and Algorithms:

1. Use Force Field to calculate the force that the substrate exerts on film. Then find the relative potential via $\vec{F} = -\nabla \cdot U$

Used empirical Forced Field to predict the structure of new interfaces in 3D space

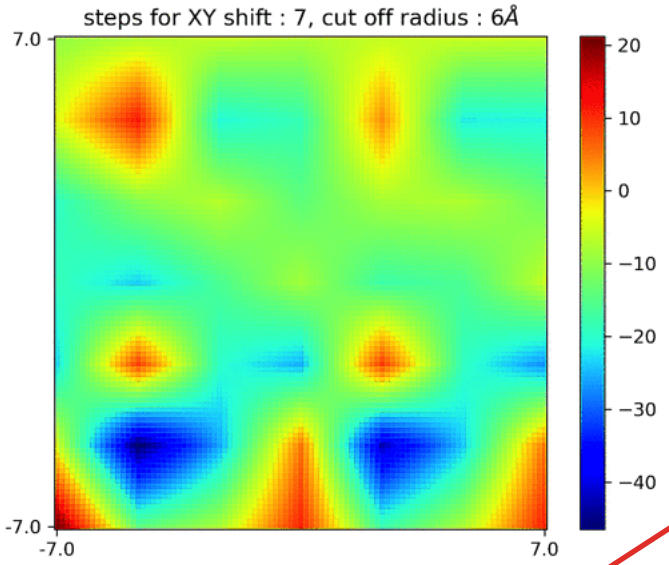
❖CLUES :

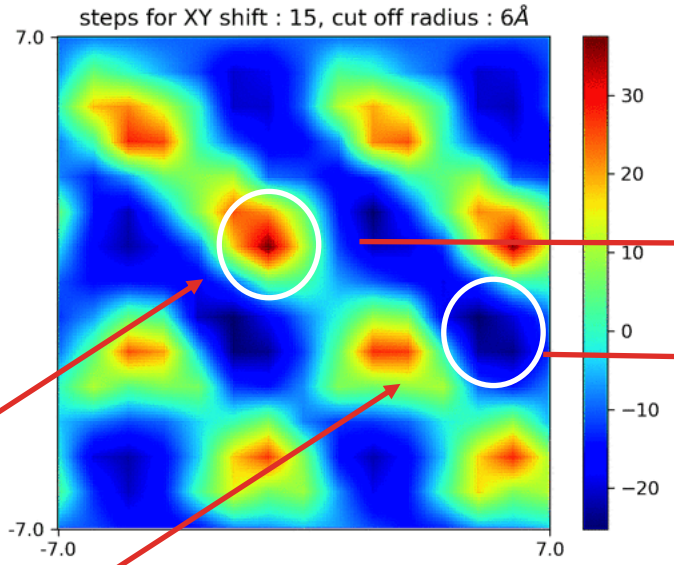
- We could easily give out the energy potential of multi-molecules system once we clearly divide interlayers into substrates and films and get the indispensable L-J parameters
- Also, we need to deal with the periodic boundary

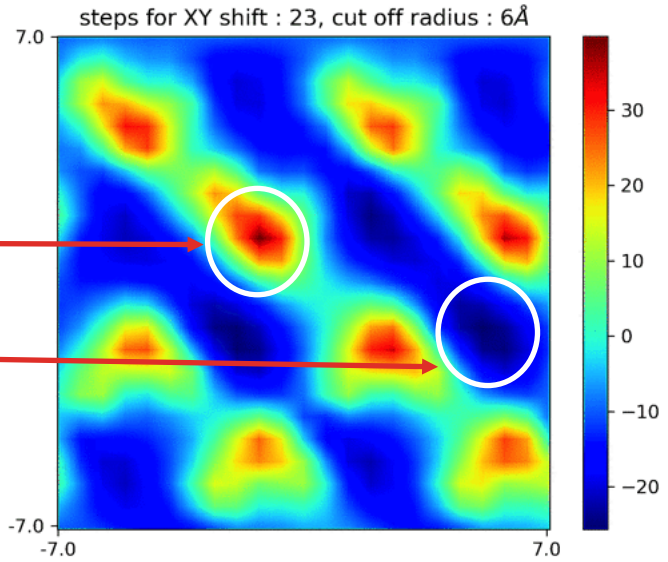


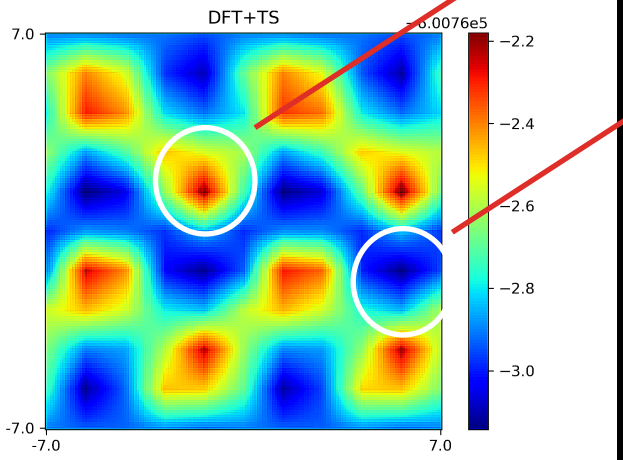
Used empirical Forced Field to predict the structure of new interfaces in 3D space

Result In X-Y direction





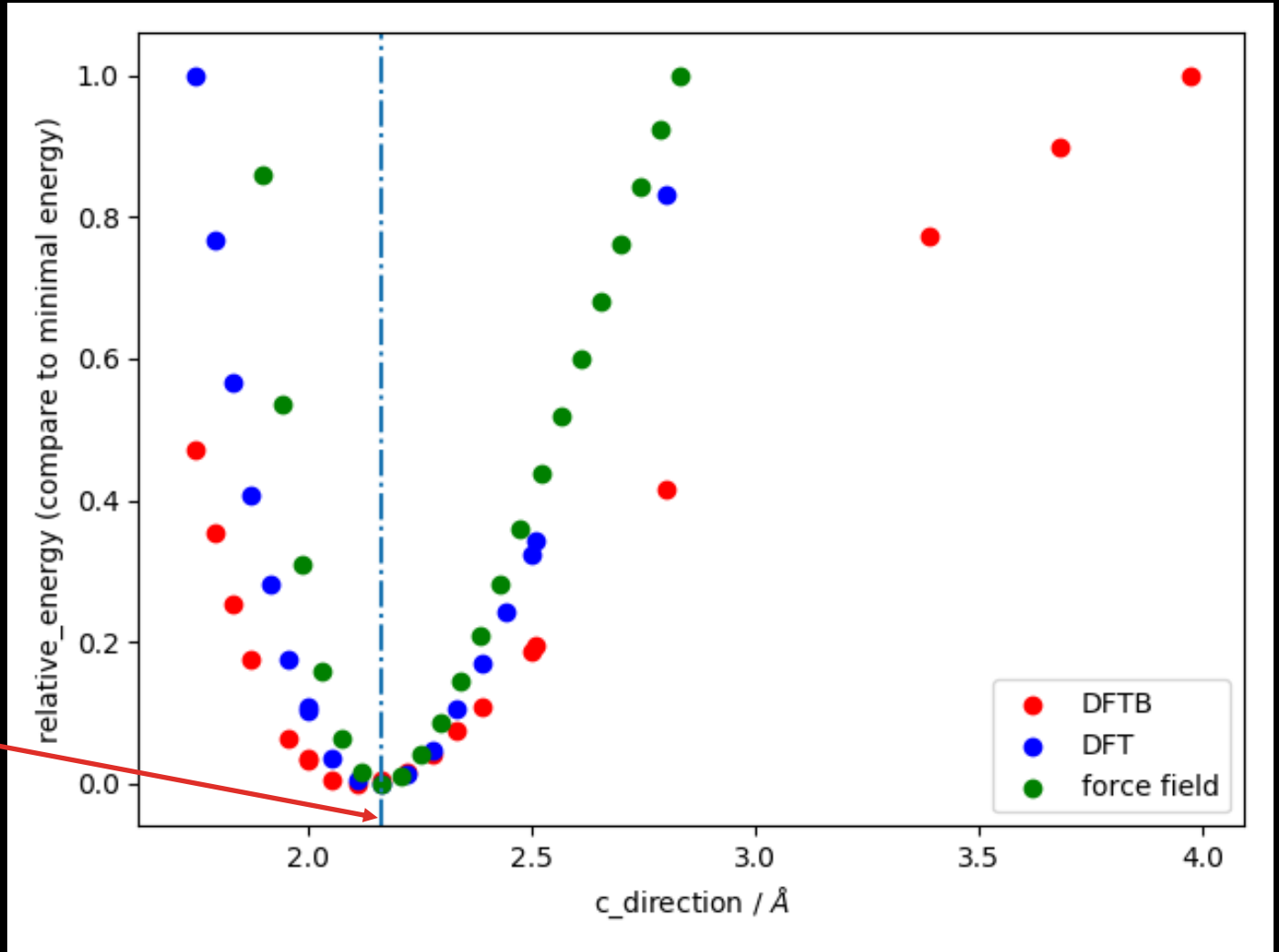




The accuracy of force field method highly depends on the moving step, which means that we need to spend more time once we want to enhance accuracy. Also, the application of force field is narrowed by limited parameters. But good news is that force field is accurate enough to match the maximal energy and minimal energy

Used empirical Forced Field to predict the structure of new interfaces in 3D space

Result in Z direction



SOME HIGHLIGHTS ON MY WORK

Used machine learning model to study materials' informatics

- Questions or goals:
- We want to solve the problems that are ubiquitous in previous empirical or semi-empirical methods, including various coordinates, non-universal potentials' expression and rare parameters.
- Methods and Algorithms:
- Used ANI-1 potentials as a reference
- Used MEGNet model as a reference

Used machine learning model to study materials' informatics
ANI-1 Potential

CLUES:

Some problems that need to be solved:

1. Training neural networks to molecules with many degrees of freedom (DOF) is difficult because the data requirements grow with each DOF to obtain a good statistical sampling of the potential energy surface.

2. The typical inputs, such as internal coordinates or coulomb matrices, lack transferability to different molecules since the input size to a neural network must remain constant.

3. The exchange of two identical atoms in a molecule must lead to the same result.

Used machine learning model to study materials' informatics

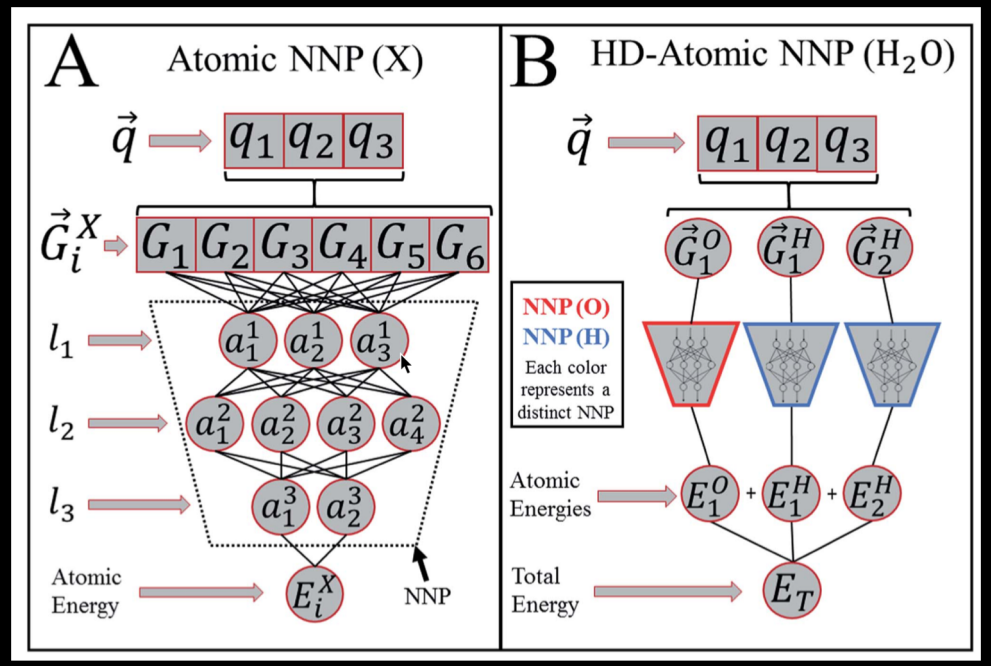
ANI-1 Potential

Base on ANAKIN-ME (ANI) model.

$$G_m^R = \sum_{j!=i}^{All\ atoms} \exp\left[-\eta\left(R_{ij} - R_s\right)^2\right] f_c(R_{ij})$$

$\vec{G}_i^x = \{G_1, G_2 .. G_M\}$composed of elements, G_M , which probe specific regions of an individual atom's radial and angular chemical environment.

$$E_T = \sum_{i}^{all\ atoms} E_i$$



ANI-1 potential changes from G_m^R to :

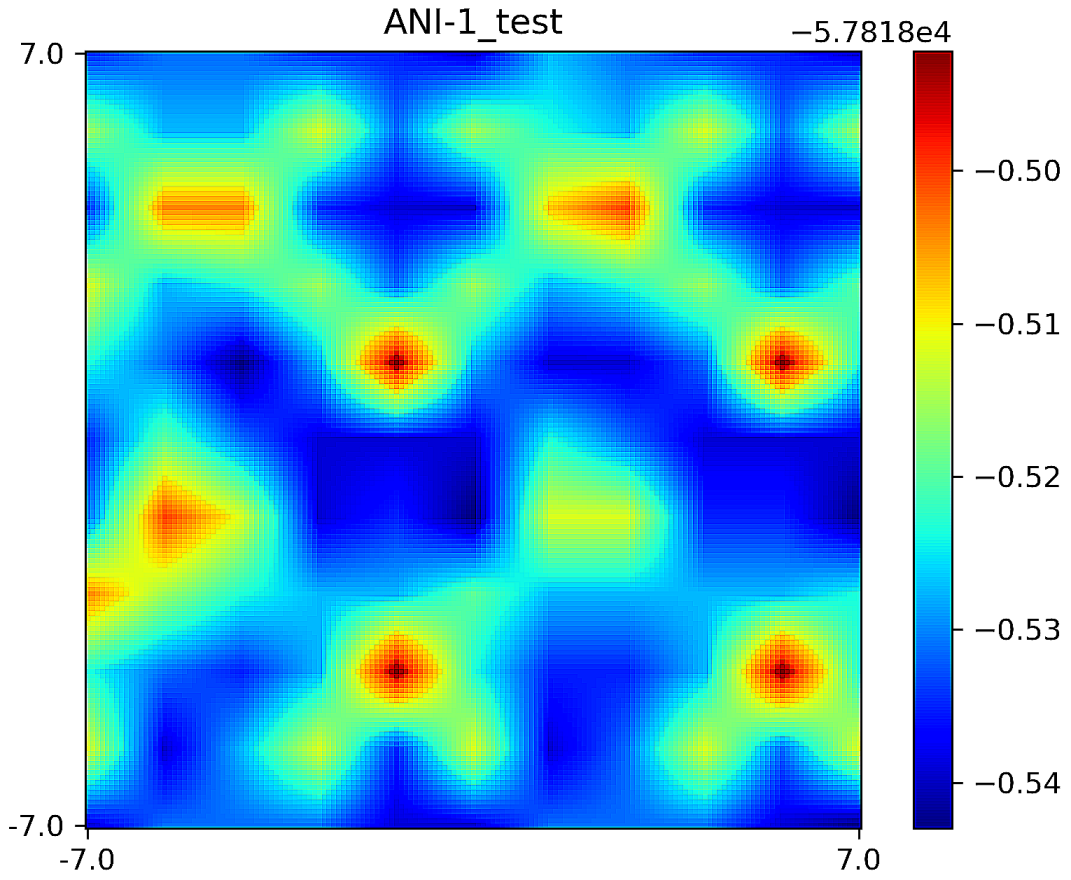
change the width of the Gaussian distribution.

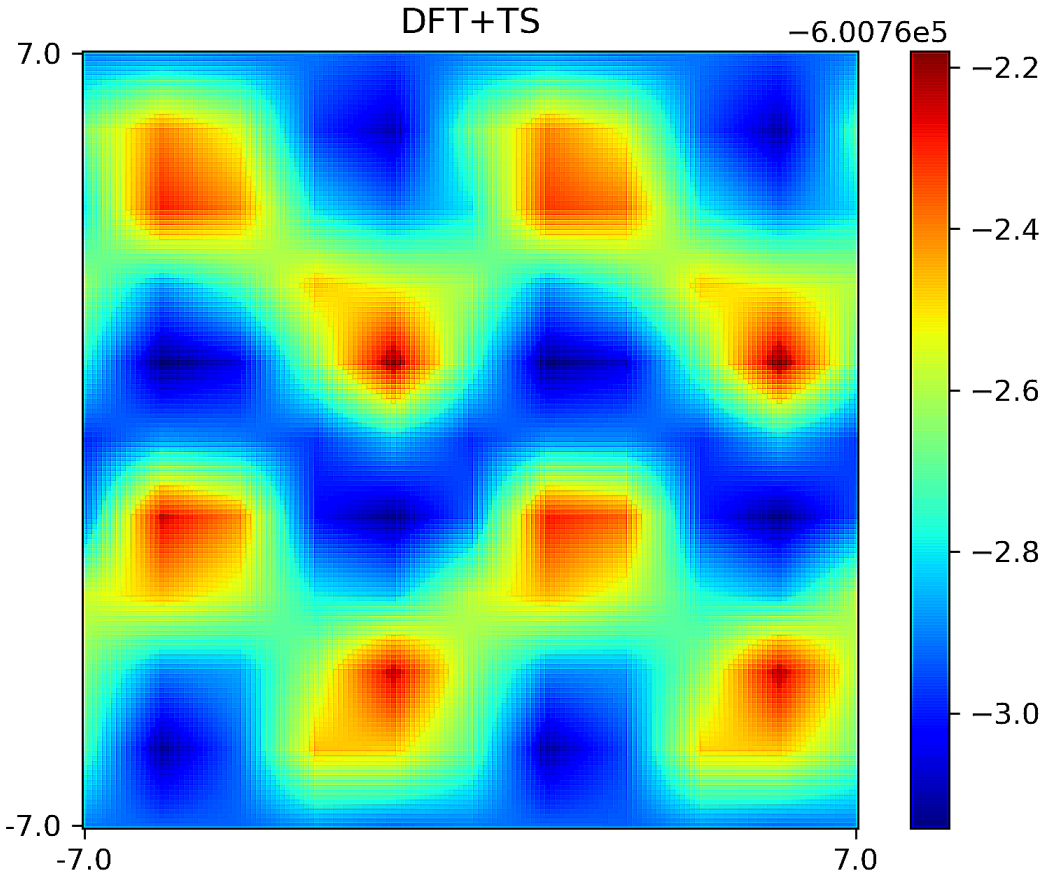
shift the center of the peak.

$$G_m^{Amod} = 2^{1-\zeta} \sum_{j,k!=i}^{All\ atoms} (1 + cos(\theta_{ijk} - \theta_s))^{\zeta} \exp\left[-\eta\left(\frac{R_{ij}+R_{jk}}{2} - R_s\right)^2\right] f_c(R_{ij})\, f_c(R_{ik})$$

Used machine learning model to study materials' informatics

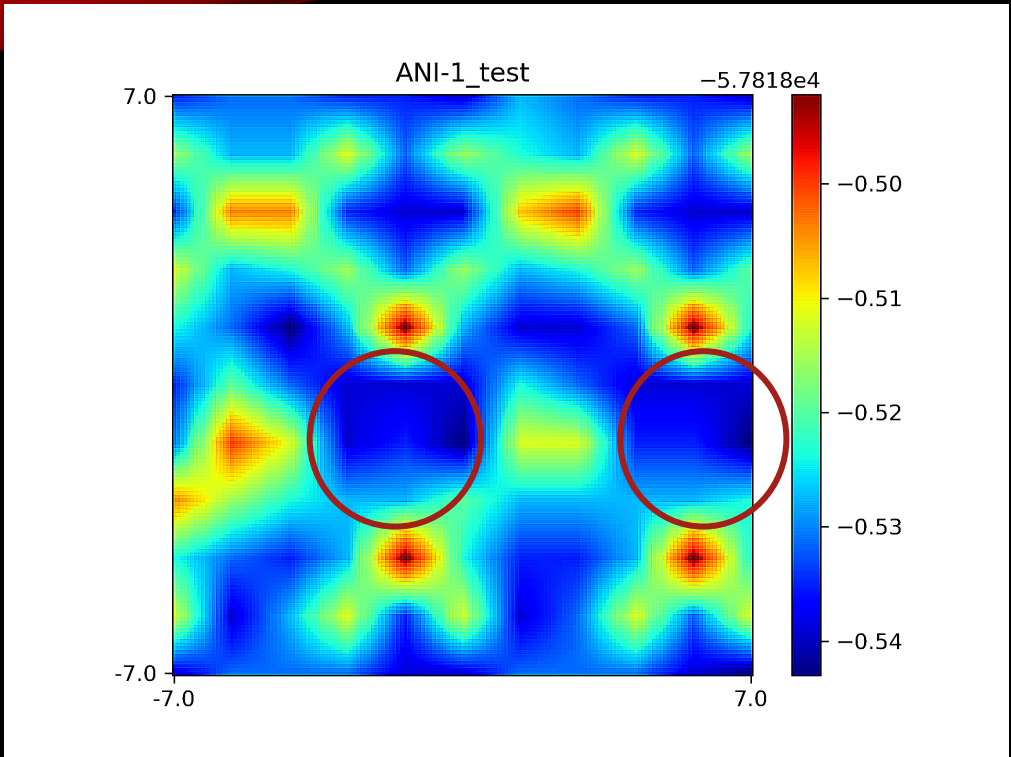
ANI-1 Potential Result

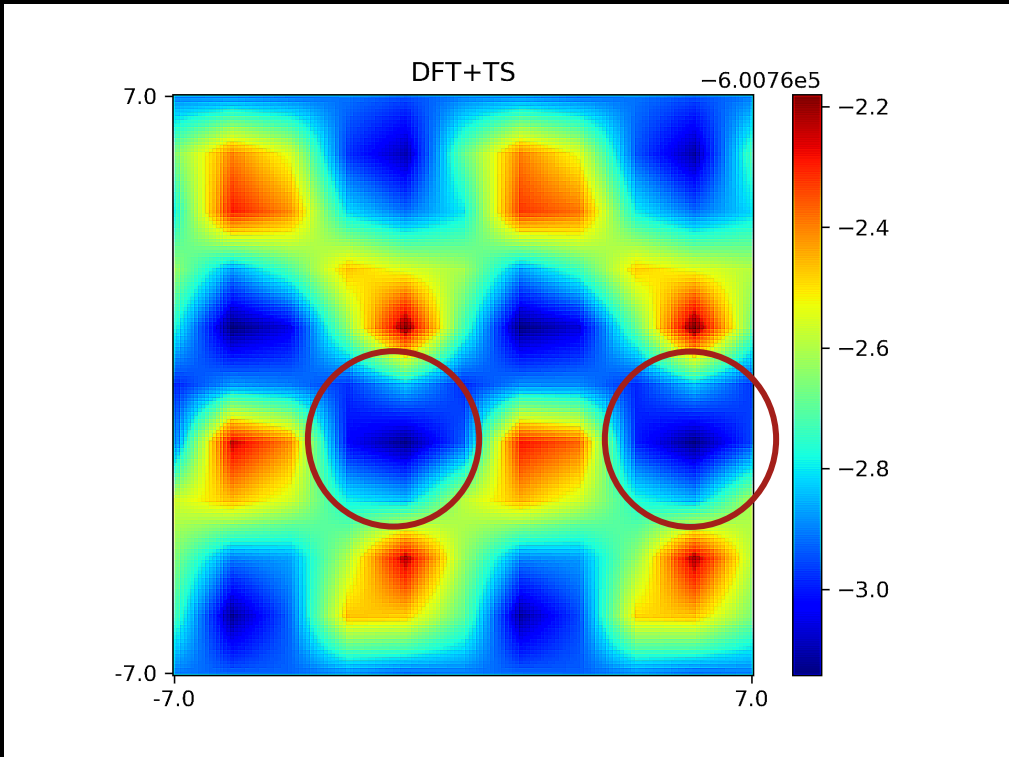




Used machine learning model to study materials' informatics

ANI-1 Potential Result





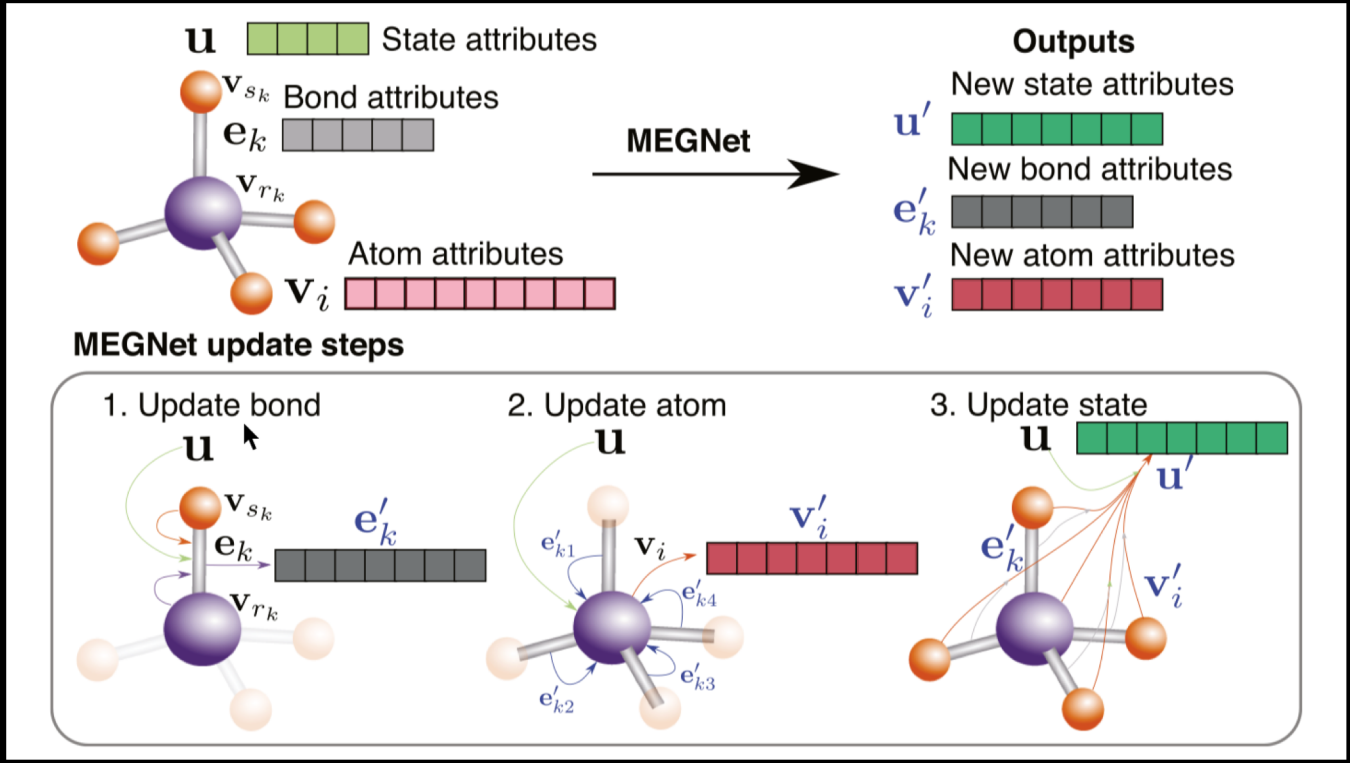
- Since the pre-trained ANI-1 potential is based on isolated molecules, it could not precisely predict the energy of multi-molecules' systems. And the predicted energy would always goes up while the distance of two molecules enlarge.

Used machine learning model to study materials' informatics
ANI-1 Potential's Outlook

- ANI-1 Potential's expression $G_m^{A_{mod}}$ naturally classify long-range and short–range interactions as one. I didn't do a test about whether this method is applicable when the training sets are more suitable since I couldn't generate lots of training sets for multi-molecules. But I do believe that the concept, generating a local environment in the following way, could be transferable in the multi-molecules' prediction.

$$G_m^{A_{mod}} = 2^{1-\zeta} \sum_{j,k!=i}^{All\ atoms} (1 + cos(\theta_{ijk} - \theta_s))^{\zeta} \exp\left[-\eta \left(\frac{R_{ij}+R_{jk}}{2} - R_s\right)^2\right] f_c(R_{ij})\, f_c(R_{ik})$$

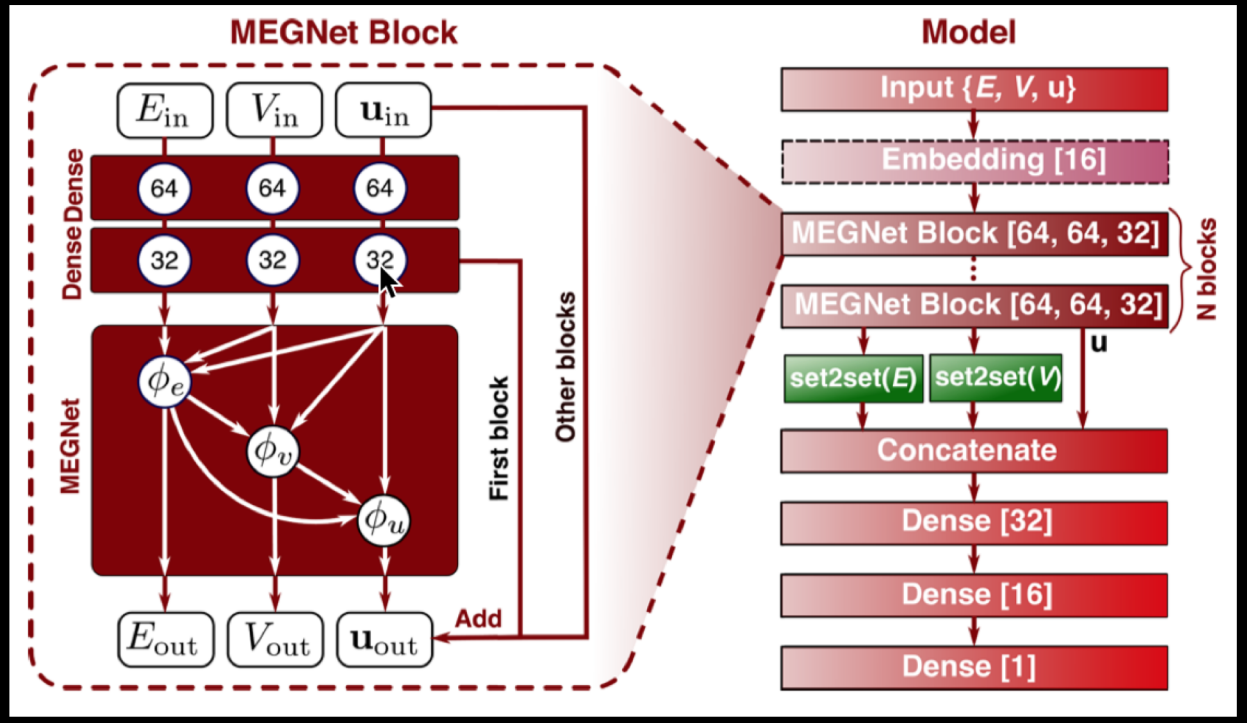
Used machine learning model to study materials' informatics
MEGNet Framework Attributes



- Let V, E, and **u** denote the atomic (vertex), bond (edge), and global state attributes.
- V is a set of V_i , which is an atomic attribute vector for atom i in a system of N atoms.
- E = $\{(e_k, r_k, s_k)\}$, e are the bonds, where e_k is the bond attribute vector for bond k, r_k and s_k are the atom indices forming bond k
- **u** is a global state vector storing the molecule/crystal level or state attributes

Used machine learning model to study materials' informatics

MEGNet Framework Update Function



$$e'_k = \phi_k(v_{s_k}, v_{r_k}, e_k, \boldsymbol{u})$$

$$\overline{v}_i^e = \frac{1}{N_i}\sum_{k=1}^{N_i} e'_k$$

$$v'_i = \phi_v(\overline{v}_i^e, v_i, \boldsymbol{u})$$

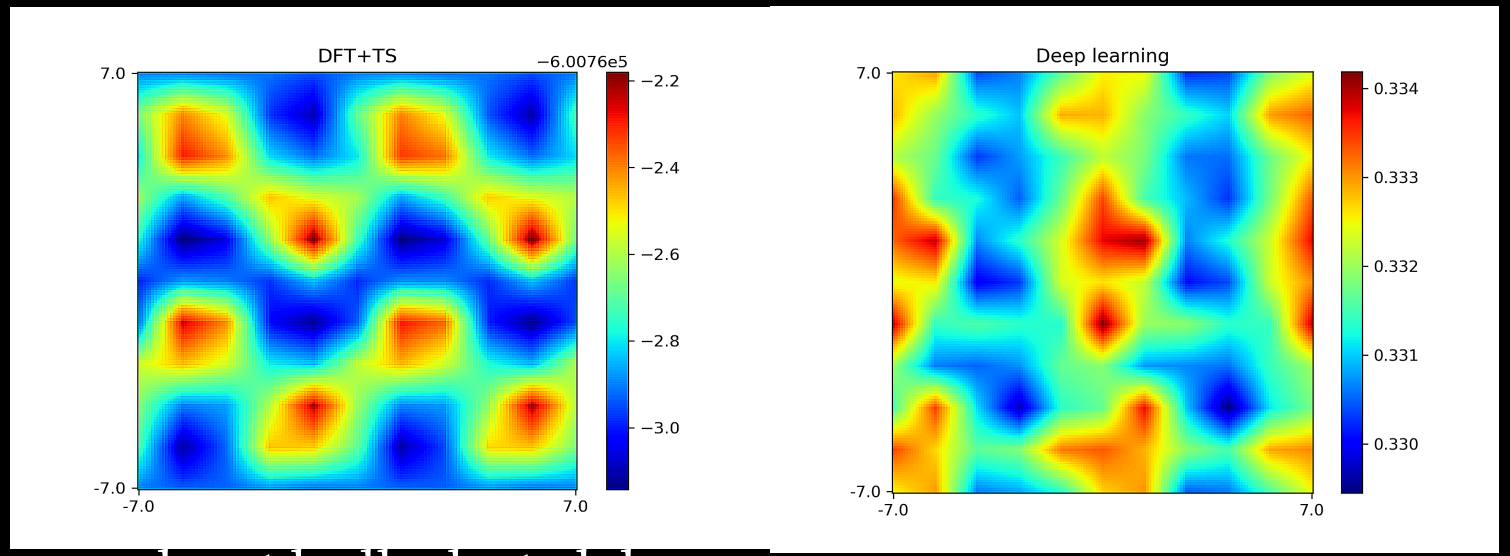
$$\overline{u}_i^e = \frac{1}{N_i}\sum_{k=1}^{N_i} e'_k$$

$$\overline{u}_i^v = \frac{1}{N_i}\sum_{k=1}^{N_i} v'_k$$

$$u'_i = \phi_u(\overline{u}_i^e, \overline{u}_i^v, , \boldsymbol{u})$$

Where $\phi(x) = W_3\left(\xi\left(W_2\left(\xi(W_1 + b_1)\right) + b_2\right)\right) + b_3$, which has two hidden layers to be universal for nonlinear functions.

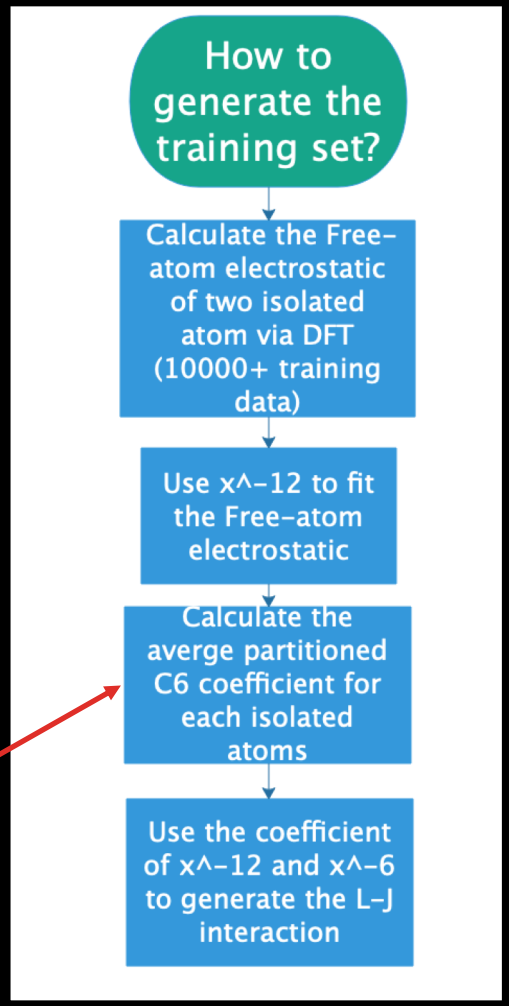
Used machine learning model to study materials' informatics
MEGNet Framework



about bulks but slabs or interlayers. So we need to generate our new training sets.

Pre-trained MEGNet performance

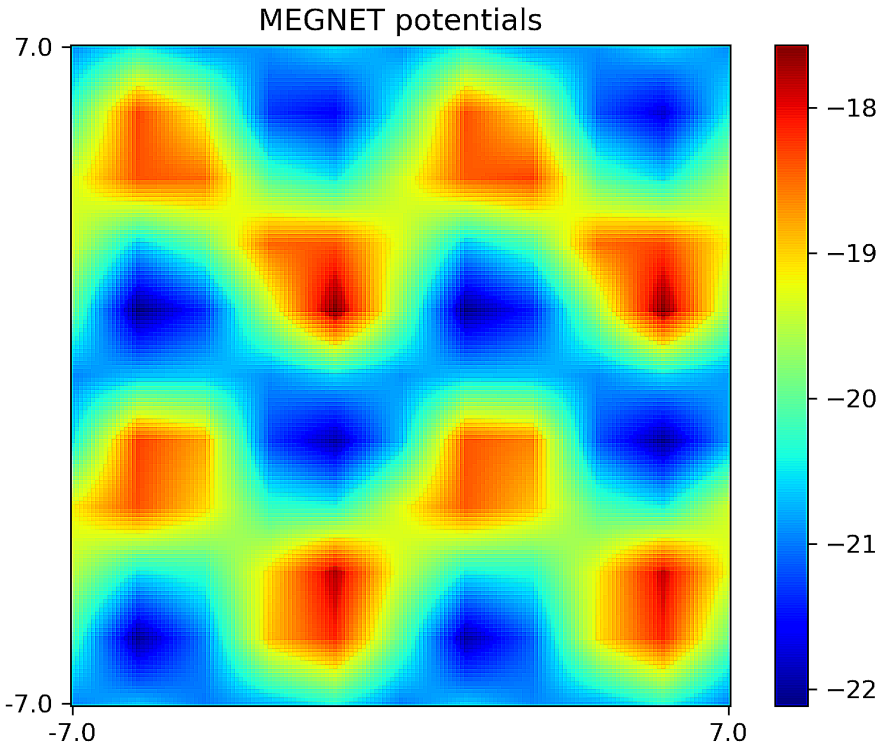
Procedure to generate the training set

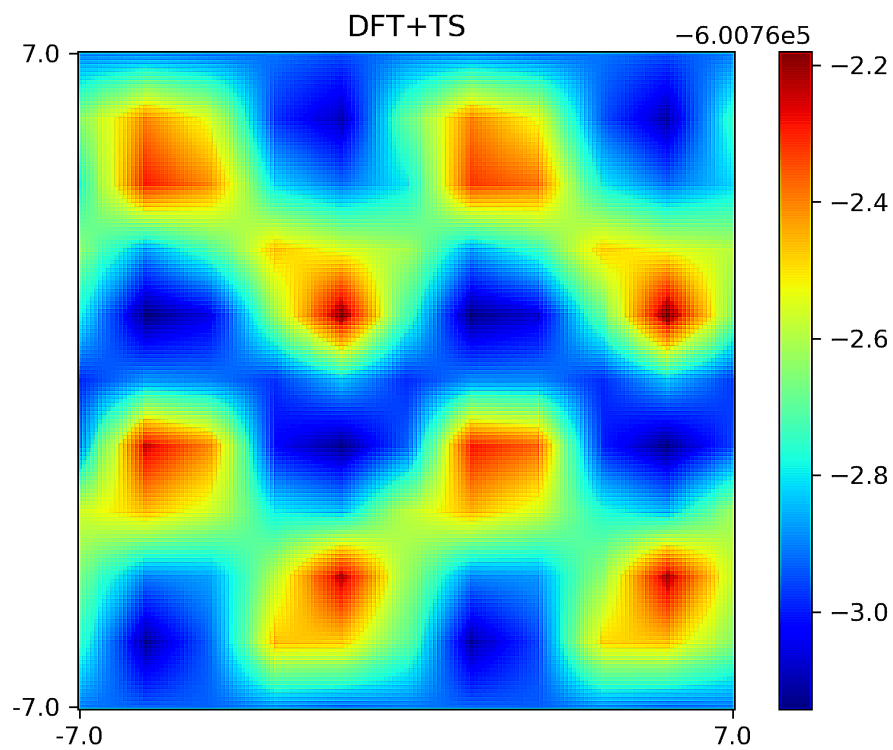


Partitioned C6 (hartree bohr^6)	alpha(bohr^3) of atom in molecule
ATOM 1 C	30.931019	9.015887
ATOM 2 H	1.870771	2.220996
ATOM 3 C	30.928487	9.015375
ATOM 4 H	1.872287	2.221936

Used machine learning model to study materials' informatics

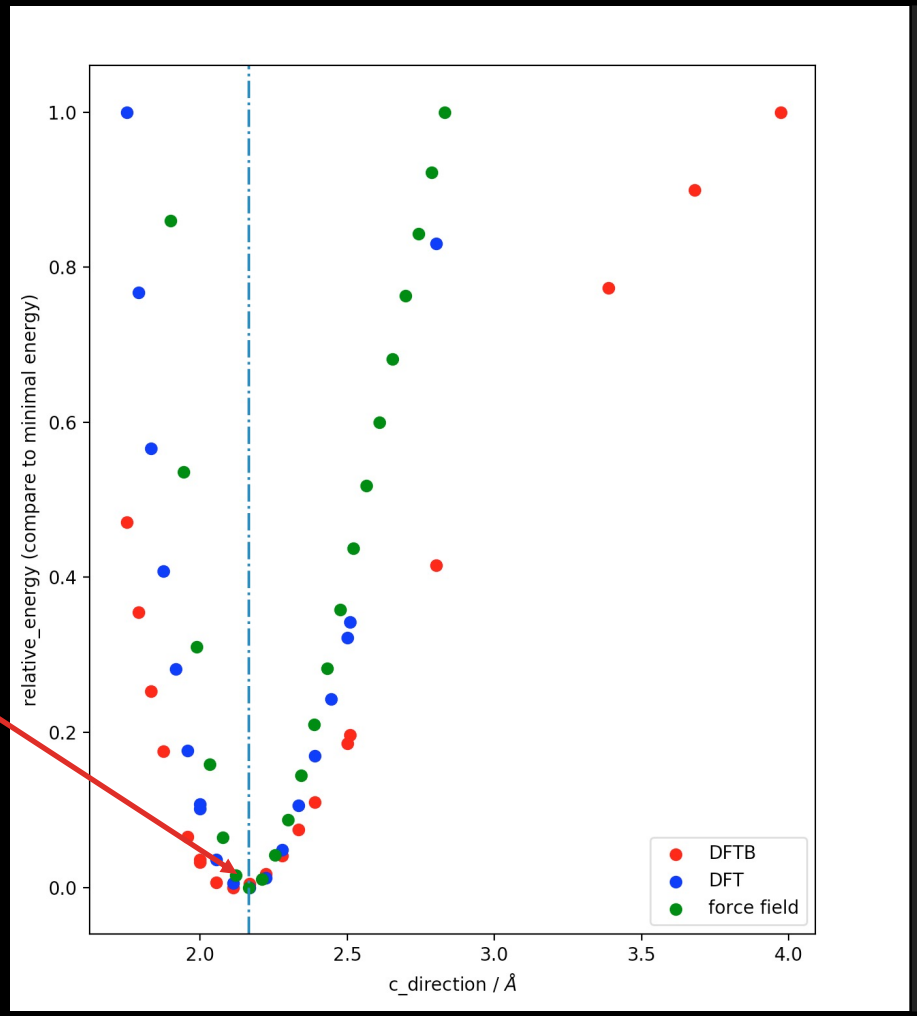
MEGNet Framework (after training) result on Rubrene-C60 System x-y direction



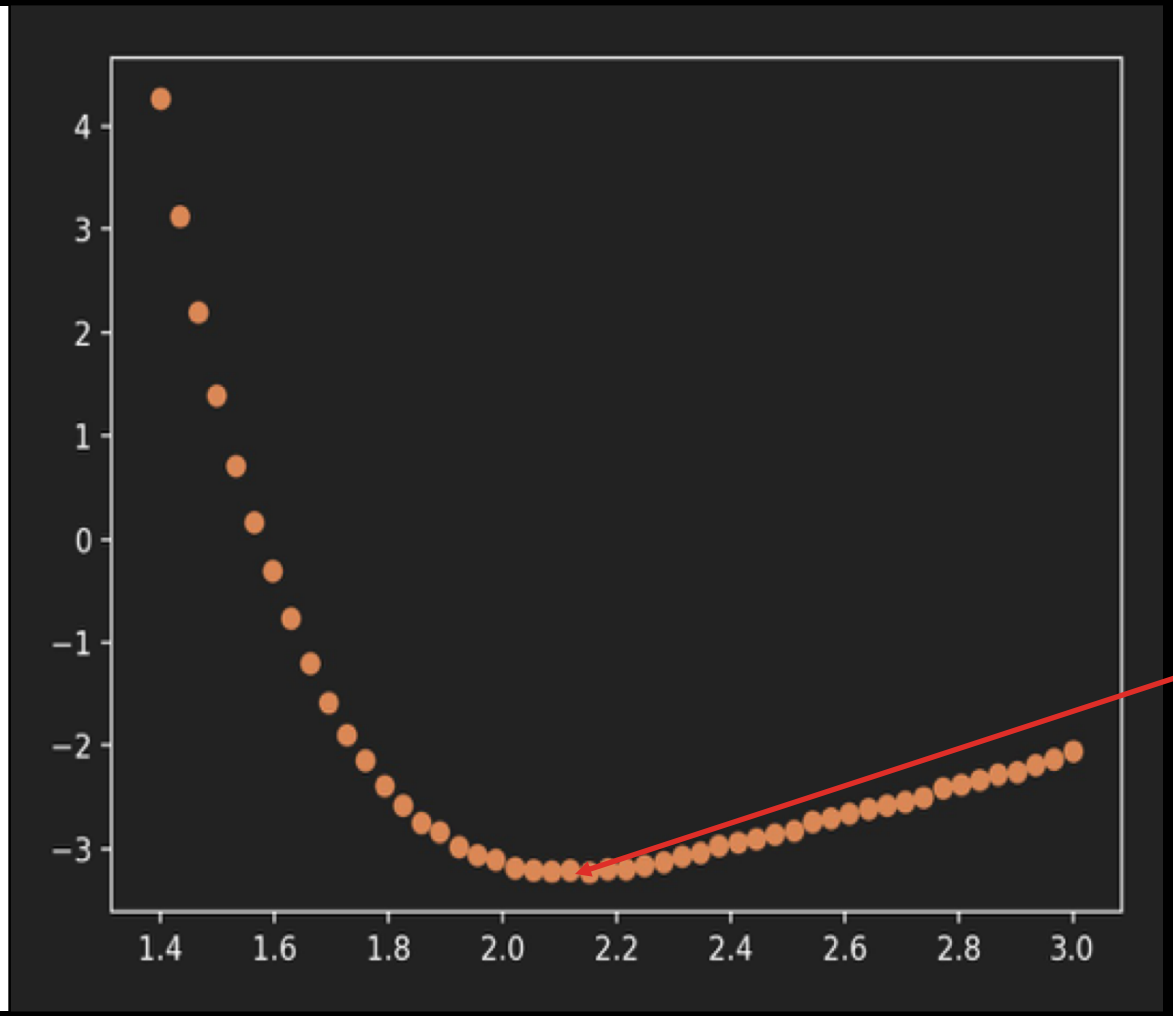


Used machine learning model to study materials' informatics

MEGNet Framework (after training) result on Rubrene-C60 System z direction



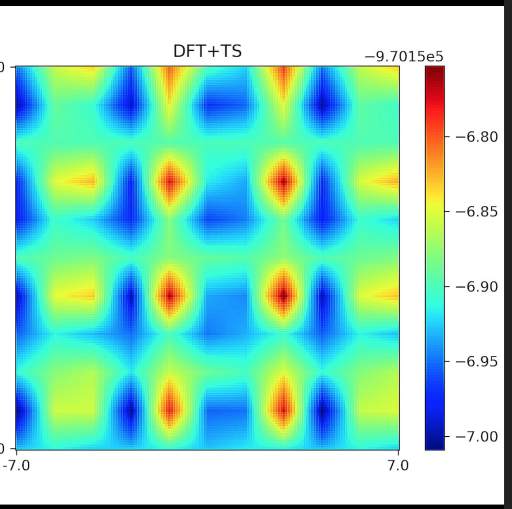
2.16

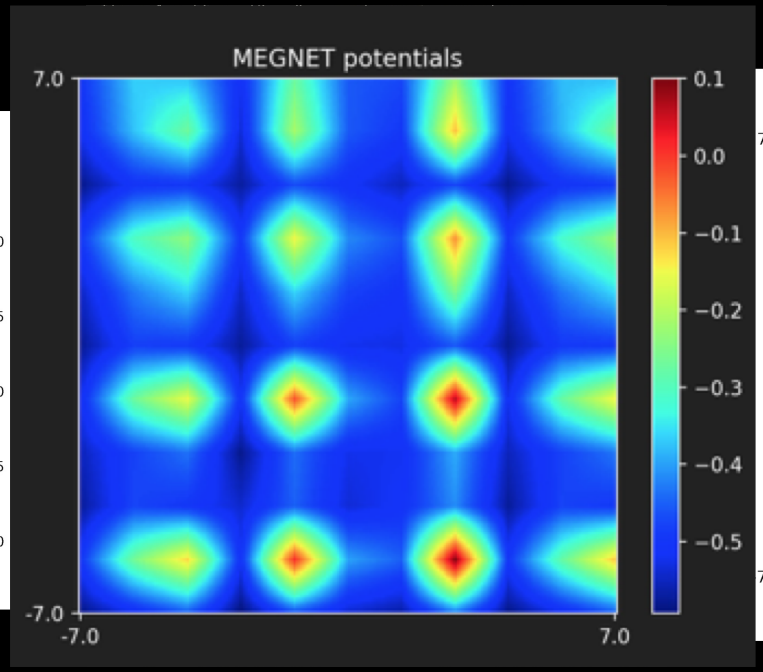


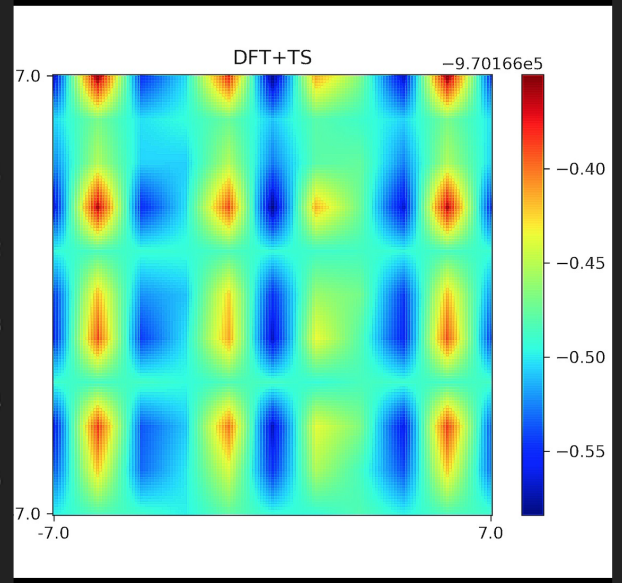
2.15

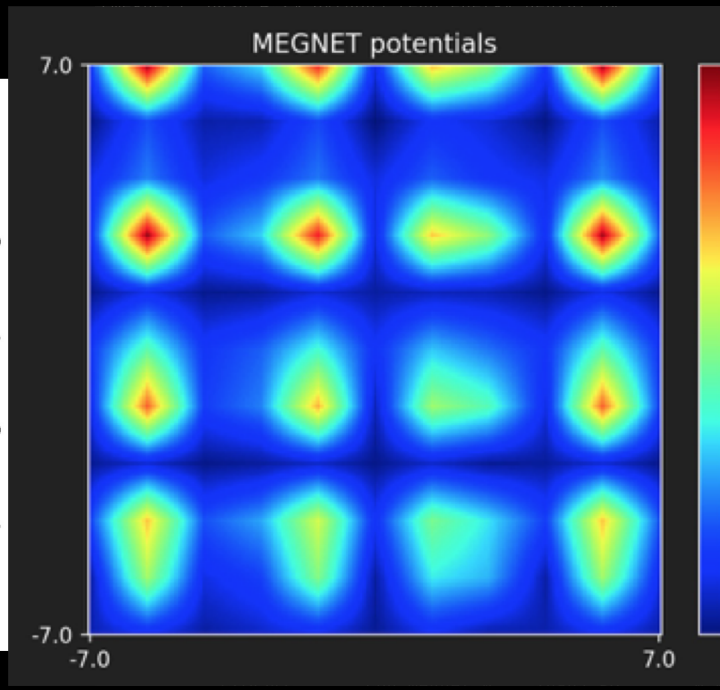
Used machine learning model to study materials' informatics

MEGNet Framework (after training) result on TTF-TCNQ System x-y direction







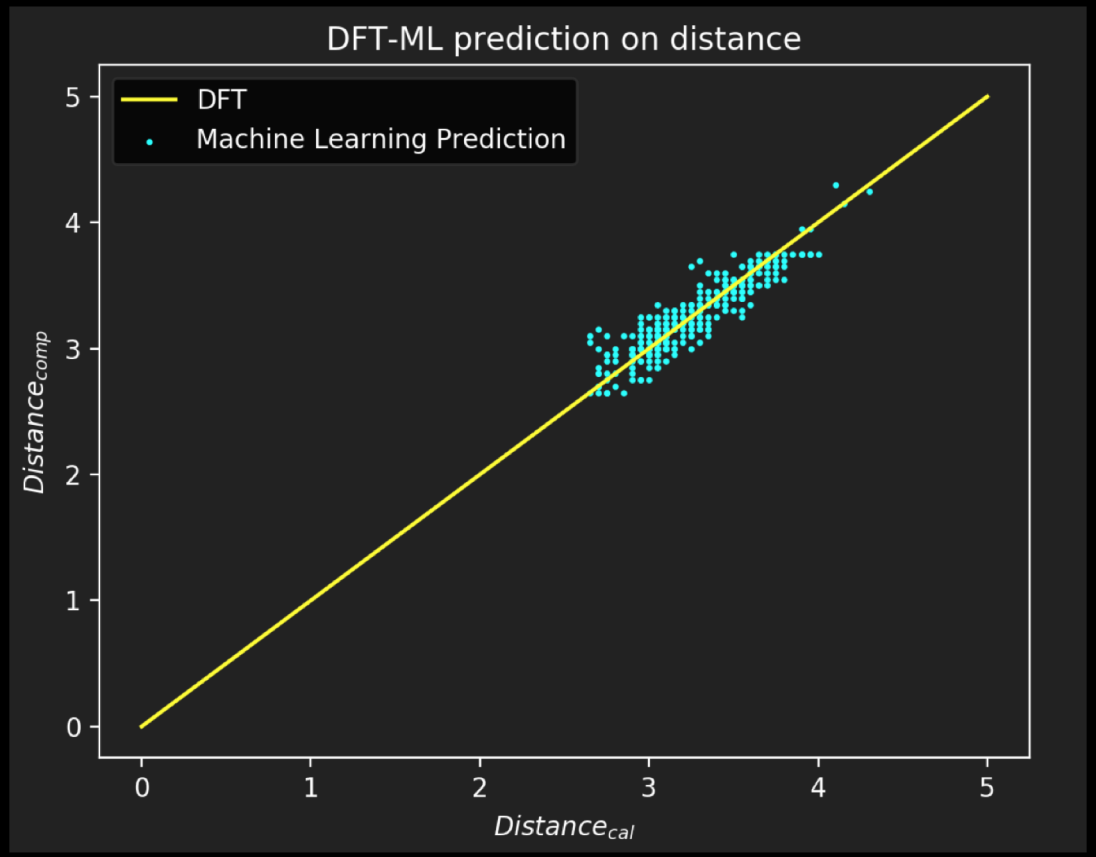


The comparisons of DFT and machine learning's results on different orientations

Used machine learning model to study materials' informatics

MEGNet Framework (after training) result on multi-molecules System

distance between two small molecules



Errors are 0.0293

500+ testing sets are tested

Used machine learning model to study materials' informatics
MEGNet framework result summary

- Pros : Errors are smaller than 3% and the efficiency is 10000+ higher that DFT once the training set is completed (10000 iterations). Also, the result shows that the MEGNet framework does produce a smooth potential, one that could provide forces, for use in molecular dynamics simulations or optimization problems.
- Cons : The way I generated the training sets is totally empirical (I need to carefully modify one parameter in this process) and highly depends on the accuracy of the results from DFT calculations. Also, the training sets just cover the non-bonded interaction between isolated atoms, which narrows its application in inorganic systems.

Used machine learning model to study materials' informatics
MEGNet framework's Outlook

- One could do MD calculations once the Monte Carlo Method is exerted on the predicted energy.
- Since the MEGNet framework adopts the similar graphic neural network as crystal graph convolutional neural networks (CGCNNs) proposed by Xie and Grossman, it should also perform well in inorganic system after a special training process.

DEVELOPMENT IN THE FUTURE

- By now, both deep learning model and the graphic algorithm are just used for basic energy calculation and molecules' identification. However, as graph-theory based algorithms, they should be able to predict more information that might exists. I do believe that the new-generation graph-theory-based algorithms could have a wide-range of application in the field of materials' informatics.

Thank you