A Forecasting System of Computational Time of DFT Calculations under the Multiverse ansatz via Machine Learning and Cheminformatics

Shuo Ma,1,2 Yingjin Ma,1,3,a) Baohua Zhang,1,3 Yingqi Tian,1,2 and Zhong Jin1,3,b)

1) Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China
2) School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 101408, China
3) Center of Scientific Computing Applications & Research, Chinese Academy of Sciences, Beijing 100190, China

(Dated: 15 November 2019)

A forecasting system for predicting computational time of density-functional theory (DFT) calculation is presented. The forecasting system is established under the many-worlds interpretation of multiverse ansatz, in which the molecules and Kohn-Sham equation are the trunk and every calculating parameters (e.g. basis set, functional) are branch points that generate result’s worlds. Every world is constituted by the solved wave functions and the accompanying data (e.g. computational time) after solving. Several machine-learning models, including random forest, long short-term memory, message passing neural network and multilevel graph convolutional network models, are employed for the prediction of computational time of any molecule belonging to a given world. For the molecules that belong to a world without pre-trained models, additional efforts are used for linear combination of models from adjacent world in order to give reasonable predictions. Benchmark results show that the forecasting system can predict proposed times with mean relative error normally around 20% when comparing to these of practical calculations.

a) Electronic mail: yingjin.ma@sccas.cn
b) Electronic mail: zjin@sccas.cn
I. INTRODUCTION

*Ab initio* electronic structure methods become more and more popular in the chemistry community, since it is recognized that the *ab initio* methods illustrate the chemical mechanism in its original view, i.e. in electronic level\(^\text{1}\text{-}6\). Normally, one needs to consider the computational costs of *ab initio* methods when determining whether they are appropriate for the problem at hand. When compared to much less accurate approaches, such as molecular mechanics, *ab initio* methods often take larger amounts of computer time, memory, and disk space. With Moore’s law\(^7,8\) of modern advances in computer science and technology such considerations are becoming less critical, but it still needs to be well considered in the foreseeable future. Because the predictable pace of the Moore’s law cannot easily cure the pain caused by the scalings in *ab initio* methods.

The HF method scales nominally as \(N^4\) (\(N\) being a relative measure of the system size, not the number of basis functions) – e.g., if you double the number of electrons and the number of basis functions (double the system size), the calculation will take 16 times as long per iteration\(^9,10\). However, in practice it can scale closer to \(N^3\) as the program can identify zero and extremely small integrals and neglect them. It can be even lower (around \(N^2\)) when a molecule contains non-ignorable linear or quasi-linear components, such as in the polypeptide case\(^11,12\). Density functional theory (DFT) methods using functionals which include Hartree–Fock exchange scale in a similar manner to Hartree–Fock but with a larger proportionality term and are thus more expensive than an equivalent Hartree–Fock calculation. DFT methods that do not include Hartree–Fock exchange can scale better than Hartree–Fock. Correlated calculations scale less favorably, though their accuracy is usually greater, which is the trade off one has to consider. For example, second-order Møller–Plesset perturbation theory (MP2)\(^13\) scales as \(N^4\) or \(N^5\), coupled cluster with singles and doubles (CCSD)\(^14,15\) scales iteratively as \(N^6\), and CCSD(T)\(^16\) scales iteratively as \(N^6\), with one noniterative step which scales as \(N^7\)\(^17\). Consequently, DFT calculations as well as its excited state counterpart, i.e. time-dependent DFT, is normally much preferred if there is no degeneracy states or no need quantitative electronic correlations.

From the statistical data in China National Grid\(^18\), more than half users choose DFT as the first choice to do the quantum chemical calculations. If the computational costs
(i.e. time, disk usage, etc.) for DFT calculations can be well predicted, it should benefit among researchers, computer centers, and public officers. For example, the researchers can better manage the normally limited computing resources, and avoid to implement some unaffordable computing tasks; the computer centers can supply elaborative expenditures before implementing the actual calculations for the users. Additionally, as we all know a rational planning could maximize the efficiency, and as such, the expectation of energy consumption can also be reduced.

The upper bound for DFT calculation is the evaluation of two-electron repulsion integrals, thus one can assume a polynomial equation that roughly uses the system magnitude feature (e.g. basis number) as independent variables $x$ and connect it with the induced variable $y$ (i.e. computational time). For example, simple $y = ax^2 + bx + c$ can be expected as the working equation for roughly predicting the time. However, simple polynomial equation is only suitable for the molecules in the same series. When molecules with different spatial structures, the predicted results are normally too poor to be referred if using this type of regression equations. Additionally, it is not convenient for these regression analysis to consider the multiple factors (e.g. basis number together with electrons, bond type, etc.), which should be coupled considered when better predictions are needed.

The exploration of elaborately predicting the computational cost and memory required for scientific programs running on high performance computers can date back to 1990s. Among these proposed methods, there are even models aiming at minimizing the environmental effect in calculating. Machine learning (ML) and artificial intelligence (AI) approaches are quite common in these models, however, these models are mainly focused on the optimization of workload scheduling. As for the field of computational chemistry, Papay et al. came up with a least square fitting method for graph-based component-wise runtime estimates in parallel self-consistent field (SCF) atomic computations in 1996. Antony et al. used a linear model to simulate the runtime of SCF algorithm in gaussian application and estimate the impacts of architectures on the basis of the count of retired instructions and cache misses. Additionally, it is worthy to be noted that Mniszewski et al. designed a class of tools for prediction of the runtime of a molecular dynamics code, allowing users to find the optimal combination of algorithmic methods and hardware options. However, as far as we know, there is
nearly no related work concerning to the prediction of computation cost in the field of quantum chemistry, except for the quantum machine learning (QML) models that very recently introduced by Heinen and coworkers. They showed that QML-based wall time predictions significantly improve job scheduling efficiency by a reduction in CPU time overhead ranging from 10% to 90%.

Inspired by many-worlds interpretation (MWI) of multiverse ansatz, we proposed a chemical MWI ansatz, in which the molecules and Kohn-Sham (KS) equation are the trunk and every calculating parameters (basis set, functional, etc.) are branch points that generate result’s worlds. Every worlds are constituted by the solved wave functions and the accompanying data (e.g. computational time) after solving. Using this ansatz, we combine the ML/AI techniques together with the cheminformatics within each result’s world, in order to develop a forecasting system for the prediction of computational time in DFT calculations whatever the models are trained or not in the given result’s world.

The paper is scheduled as following: In Sec.II, we present brief descriptions of A) the DFT theory and its scaling, B) the main idea of chemical MWI ansatz, cheminformatics, C) basic ML/AI techniques for the purpose of predictions, D) combination of chemical MWI, cheminformatics and ML/AI as well as, E) workflow of the established forecasting system. Benchmark examples are presented in Sec.IV with computational details are explained in Sec.III. Finally, we draw our conclusions in Sec.V.

II. THEORY AND METHODOLOGY

A. Kohn-Sham density functional theory and its scaling

In the Born-Oppenheimer approximation, a stationary electronic state can be described by a wave function $\Psi(\vec{r}_1, ..., \vec{r}_N)$ satisfying the many-electron time-independent Schrödinger equation,

$$\hat{H}\Psi = [\hat{T} + \hat{V} + \hat{U}]\Psi = \left[\sum_{i=1}^{N} \left(-\frac{\hbar^2}{2m_i}\nabla_i^2\right) + \sum_{i=1}^{N} V(\vec{r}_i) + \sum_{i<j}^{N} U(\vec{r}_i, \vec{r}_j)\right] \Psi = E\Psi \quad (1)$$

where the $\hat{H}$ is the electronic Hamiltonian, $E$ is the total energy, $\hat{T}$ is the kinetic energy, $\hat{V}$ is the potential energy from the external field due to positively charged nuclei, and $\hat{U}$ is the electron-electron interaction energy.
In the KS-DFT hypothesis, particles can be treated as non-interacting fermions, so that there exists an orthogonal and normalized function set \( \{ \phi^K_S | i = 1, 2, \cdots, N \} \) satisfying the condition:

\[
\rho(\vec{r}) = \rho_s(\vec{r}) = \sum_{i=1}^{N} |\phi^K_S(\vec{r})|^2
\]  

(2)

\( \rho(\vec{r}) \) is the probability density of ground state electrons in a factual system and \( \rho_s(\vec{r}) \) is that in a fictitious system. As such, the KS wave function is a single Slater determinant constructed from a set of function sets (i.e. orbitals) that are the lowest energy solutions to

\[
\left[ -\frac{\hbar^2}{2m_1} \nabla^2_1 - \sum_A Z_A \frac{1}{r_{1A}} + \int \frac{\rho(\vec{r}_2)}{r_{12}} d\vec{r}_2 + V_{XC}(1) \right] \phi^K_S(1) = \epsilon^K_S \phi^K_S(1),
\]

(3)

where the \( V_{XC}(1) \) is called exchange-correlation potential, and the ”(1)” following each operator symbol simply indicates that the operator is 1-electron in nature. This equation is very similar to the Fock equation,

\[
\left[ -\frac{\hbar^2}{2m_1} \nabla^2_1 - \sum_A Z_A \frac{1}{r_{1A}} - \frac{1}{2} \int \int \frac{\rho(\vec{r}_1)\rho(\vec{r}_2)}{r_{12}} d\vec{r}_1 d\vec{r}_2 \right] \phi^{HF}(1) = \epsilon^{HF} \phi^{HF}(1),
\]

(4)

that used in Hartree-Fock theory. Both Eq.(3) and Eq.(4) can be solved iteratively using the so-called self-consistent field (SCF) methods. During the SCF iteration, orbital \( \phi_i \) is updated iteratively, and is used to calculate electronic density \( \rho \),

\[
\rho(\vec{r}) = \sum_{i=1}^{N} |\phi_i(\vec{r})|^2
\]

(5)

which in turn correct the 1-electron matrix (e.g. Fock matrix in SCF iterations) that to be diagonalized. After several iterations, both molecular orbital \( \phi_i \) and its energy \( \epsilon_i \) can be obtained, then the total electronic energy can be calculated.

Comparing Eq.(3) with Eq.(4), one can clearly see that the major difference between them is in the 2-electron integrals part. The origin of the \( N^4 \) scaling behaviour is the calculation of four-center 2-electron integrals, i.e.

\[
(\mu\nu|\lambda\sigma) = \int \int \phi_{\mu}(1)\phi_{\nu}(1) \frac{1}{r_{12}} \phi_{\lambda}(2)\phi_{\sigma}(2) d\tau_1 d\tau_2,
\]

(6)

where \( \mu, \nu, \lambda, \) and \( \sigma \) are indices of atomic orbitals. This scaling is also the upper boundary for the Hartree-Fock or hybrid DFT calculations. However, many 2-electron integrals are of negligible magnitude for large molecules, as well as some rigorous upper boundary
conditions can be employed to the integrals. For example, the Schwarz inequality \[ |(\mu\nu|\lambda\sigma)| \leq \sqrt{(\mu\nu|\mu\nu)(\lambda\sigma|\lambda\sigma)} \] allows the calculation of strict, mathematical upper bounds to all 2-electron integrals to be computed in an \( N^2 \log N \) process. Aside from the calculation of the 2-electron integrals, the diagonalization of the Fock or Fock-like matrix is expected to contribute significantly. Because the diagonalization step scales intrinsically as \( N^3 \), or even lower if sufficiently sparse matrix (e.g. in large enough molecule) to be diagonalized.

Nevertheless, one should notice that for the hybrid DFT functionals, a hybrid exchange-correlation functional (i.e. \( V_{XC}(1) \) term in Eq.(3)) is usually constructed as a linear combination of the third terms (Hartree-Fock exact exchange functional) in Eq.(4). Hence, hybrid DFT methods scale in a similar manner to Hartree-Fock but normally more expensive due to a larger proportionality term, while the pure DFT methods scale better than Hartree-Fock because there is no Hartree-Fock exchange.

B. Chemical MWI ansatz and cheminformatics

The chemical MWI ansatz used in this paper is inspired by Hugh Everett III’s many-worlds interpretation (MWI) of multiverse, in which an interpretation of quantum mechanics that asserts that the universal wave function is objectively real, and that there is no wavefunction collapse. It implies that all possible outcomes of quantum measurements are physically realized in some “world” or universe, and they all share a unique start-point. In our case, the unique start-point is the molecules to be calculated together with KS equation (i.e. Eq.(3)), and every result’s world contains its own parameters (e.g. basis set, functional, convergence parameters, etc.), wave functions, as well as the computational times for molecules, as illustrated in the FIG.7.

It is noteworthy that all universes share a same start-point, i.e. molecules, thus it is critical important for extracting the connections among molecules. The cheminformatics, which combines the areas of topology, chemical graph theory, information retrieval and data mining in the chemical space, is used for the management and analysis of chemical information. Such information can be of a variety of types including chemical structure (in various formats such as smiles molecular input line entry specification (SMILES),
FIG. 1. The illustration of origional MWI and the chemical MWI ansatz

structure data format (SDF\textsuperscript{37}, and so on) and derived aspects of chemical structure (such as number of atoms and various descriptors of structure). Furthermore, the information that obtained from cheminformatics can be used in the ML/AI models to develop a forecasting system, so that it can bypass the actual calculations. The details of ML/AI models that combine cheminformatics and MWI ansatz are described in the following subsections.

C. Prediction of computational time by ML/AI and cheminformatics

In a given DFT world, we developed several ML/AI models from simple to complex, in order to give a reliable prediction for the computational time (herein, the computational time means the cpu time). The four ML/AI models are random forest (RF)\textsuperscript{38}, long short-term memory (LSTM)\textsuperscript{39}, message passing neural network (MPNN)\textsuperscript{40}, and multi-level graph convolutional neural network (MGCN)\textsuperscript{41}, respectively. The ideas behind them are structural similarity (in RF), chemical formula recognition (in LSTM), and graph based predictions (in MPNN and MGCN).

1. RF model together with simple feed-forward neural networks

We use the feed-forward neural network (FNN) as the skeletal frame to obtain the model between basis number and the computational time. The illustration of the FNN model is shown in FIG. 2. Four layers are used in our model, which are input layer, two hidden layers, and output layer. The “input layer” are constructed by the system magnitude features (e.g. number of basis sets) and the computational time. These vectors
are normalized and then fed to the hidden layers. Each “hidden layer” contains several neurons, and the TANH function is used as the activation function. The data passed from the hidden layer will be directly used to linear combination for the output results.

![Simple FNN model](image)

**FIG. 2.** The simple FNN model

If we only use the FNN model of FIG. 2, the predicted results for an molecule that far from the training sets will still be very poor. In order to overcome the dependency of training sets, we introduced the idea of “feature training”. The “feature training” means few molecules suites with specific features (e.g. linear(L), dendritic(D), ring(R) etc.) will be trained as the “cost functions”, upon which the computational cost \( y \) for any molecule can be calculated by the linear combination of these “cost basis”, e.g.

\[
y = p_L \cdot f_L(x) + p_D \cdot f_D(x) + p_R \cdot f_R(x) + ..., (8)
\]

where the \( p_L, p_D, p_R \) are the possibilities for each “cost basis” \( f_{feature}(x) \), the \( f_{feature} \) is the ”cost functions”, and the \( f_{feature}(x) \) denotes the expected computational cost for the “feature” model with magnitude parameter \( x \). Herein, one can notice that this ”feature training” ansatz matches well with the RF model that used in ML/AI. Under this ansatz, a specific model (i.e. cost function) will be trained and saved for each type of structure molecular suit. After that, an RF classifier is used to classify the molecules into given categories, such as the linear, dendritic or ring molecules, etc., with possibilities (\( p \)). The classifier accepts the SMILES codes of molecules as its input. The number of every molecule’s atoms (with hydrogen atoms excluded), branches, atoms on branches and cycles will be calculated and combined as an input vector according to its SMILES
code. The entire process is illustrated by FIG.3

2. Bi-LSTM with attention

The kinds of features extracted by RF classifier are designed artificially with subjective preference, so that it may be not enough for aggregating molecular structural information. Considering that we use textual data (i.e. SMILES code) as the representation of molecular structure, methods for natural language processing (NLP) is suitable for feature extraction in this issue.

Here, we use the bidirectional LSTM(Bi-LSTM) with attention model that proposed by Peng et al.. It is the state-of-art model for relation classification tasks, and as such, we utilize this model to extract structural features from SMILES. The architecture of the model is illustrated in FIG.4. The input features include the SMILES code (in the form of one-hot) and the number of basis functions. Assuming every character in a T-length SIMLES sequence is denoted with a one-hot vector $x_i$, $x_i$ will be converted to a real number vector $e_i$ ($e_i = Wx_i$, $W$ is a parameter matrix pre-trained with word2vec algorithm) at the word embedding phase. Then $E = \{e_1, e_2, \cdots, e_t\}$ is sent to the Bi-LSTM layer. Bi-LSTM layer consists of a forward layer and a backward layer so that the model can learn from forward and backward sequences since the past and future semantic information in a sentence are equivalently significant. The LSTM layers at two directions generate two outputs, $H_f$ and
\[ H_b: \]

\[ H_f = L(E), \quad H_b = L(E) \]  \hspace{1cm} (9)

where \( L \) denotes the operations performed by a LSTM layer. Then an attention layer accepts the sum of the outputs from Bi-LSTM layers:

\[ H = H_f + H_b \]  \hspace{1cm} (10)

Attention mechanism allows different context vectors generated from Bi-LSTM layer’s output at every time step by assigning different “attention weight” to the outputs. Without attention, the feature extraction operation on the output at every time step will have the limitation of depending on one same context vector with fixed length invariant to time steps. Attention layer outputs the final representation \( c \) of a SMILES as

\[ \alpha = \text{softmax}(w^T \tanh(H)) \]  \hspace{1cm} (11)

\[ c = Ha^T \]  \hspace{1cm} (12)

where \( \alpha \) is the attention weight vector, and \( w \) is a trained parameter vector. The high-level features of molecular structures are produced after the attention layer. We combine the structural features and the number of basis functions and feed them into full connected layers to get the predictive result.
3. **MPNN model**

As a representation of molecular structure, SMILES is quite rough due to absence of spatial information. For more accurate representation, it is rational to model a molecule using an undirected graph $G$ (shown in FIG.5). We use the MPNN model, which is recognized as a kind of GNN (Graph Neural Network), proposed by Glimer as a solution for graph-based learning.

The initial inputs of the model include a feature vector collection for nodes of the graph, denoted with $x_v$, containing features of atom types, aromaticity and hybridization types, and a feature vector collection for edges, denoted with $e_{vw}$, containing features of bond types. Then there exist two phases, a message passing phase and a readout phase. The message passing phase totally runs $T$-step graph convolutions and at each step $t$, it is defined in terms of a message function $M_t$ and a vertex update function $U_t$. Before the message passing, the node vectors are mapped to a $n \times d$ matrix called ”node embedding” by a network (called ”Node net”), with $n$ the number of nodes, $d$ the dimension of hidden state of each node. During the message passing phase, hidden states $h^t_v$ of each node are updated according to messages $m^{t+1}_v$. So the message passing phase can be summarized as

$$m^{t+1}_v = \sum_{w \in N(v)} M_t(h^t_v, h^t_w, e_{vw})$$

$$h^{t+1}_v = U_t(h^t_v, m^{t+1}_v)$$

where $N(v)$ denotes the neighbors of $v$ in $G$. $M_t$ is defined as $M(h_v, h_w, e_{vw}) = A(e_{vw})h_w$ specifically, where $A(e_{vw})$ is a network (Edge net) mapping each edge vector $e_{vw}$ to a $d \times d$ matrix (edge embedding). The vertex update function is GRU, short for gated recurrent unit. At the readout phase, a feature vector can be gotten as a summary of the whole graph with a readout function $R$

$$\hat{y} = R(\{h^T_v | v \in G\})$$

where $R$ is the set2set model. The set2set model produces a graph level embedding which is invariant to the order of nodes. Finally, we combine the graph level embedding and the basis function number and feed them to full connected networks to get the predictive results. The architecture of the MPNN model is illustrated in FIG.5.
4. **MGCN model**

Apart from MPNN, we also introduced another graph neural network model, MGCN [41], which is reported to have the advantages of generalizability and transferability. As illustrated in FIG. 6, the architecture of MGCN includes 5 phases. First of all, the initial inputs of the model include a feature vector collection for nodes of the graph, denoted with \( a^0 \in \mathbb{R}^D \), containing features of atom types, aromaticity and hybridization types, a feature vector collection for edges, denoted with \( e \in \mathbb{R} \), containing features of bond types and bond lengths, and the number of the basis functions. At the pre-processing phase, the embedding layer generates the node atom embeddings \( (A^0 \in \mathbb{R}^{N \times D}) \) and edge embeddings \( (E \in \mathbb{R}^{N \times N \times D}) \). The radial basis function (RBF) [46][47] layer converts...
the bond lengths to a distance tensor $D \in \mathbb{R}^{N \times N \times K}$, with $d_{ij}$ representing the distance between atom $i$ and atom $j$. The RBF layer’s function form is elaborated as

$$RBF(x) = \frac{K}{i=1} h(||x - \mu_i||),$$

(16)

where $\sim$ denotes concatenation and $\mu_i$ is from a set of $K$ central points $\{\mu_1, \ldots, \mu_K\}$. At the message passing phase, the interaction layers are constructed in the form of hierarchical architecture to simulate the quantum interactions which are transformed at different levels (atom-wise, atom-pair, atom-triple, …). The $l$-th layer generates an edge representation $e_{ij}^{l+1}$ and an atom representation $a_i^{l+1}$:

$$e_{ij}^{l+1} = h_e(a_i^l, a_j^l, e_{ij}^l),$$

(17)

$$a_i^{l+1} = \sum_{j=1, j\neq i}^{N} h_v(a_j^l, e_{ij}^l, d_{ij}),$$

(18)

where $h_e$ is the edge update function and $h_v$ is the message passing function, respectively. The form of $h_e$ is:

$$h_e = \eta e_{ij}^l \oplus (1 - \eta) W_u^e a_i^l \odot a_j^l.$$  

(19)

Here $\eta$ is a constant set to 0.8, $W_u^e$ is a weight matrix, $\oplus$ denotes the element-wise plus and $\odot$ is the element-wise dot product, respectively. The form of $h_v$ is:

$$h_v = \tanh(W_v^u (M^a(a_i^l) \odot M^d(d_{ij}) \oplus M^e(e_{ij}))),$$

(20)

where $M(x)$ means a linear layer which is in the form of $M(x) = Wx + b$. The outputs of $T$ interaction layers along with $a_i^0$ are concatenated together as

$$a_i = \sum_{k=0}^{T} a_i^k.$$  

(21)

After that, The Read-Out layer generates a graph-level embedding $G$ as

$$G = \sum_{i=1}^{N} W_u^g \sigma(M^g(a_i)) + \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} W_r^e \sigma(M^e(e_{ij})).$$

(22)

Here $\sigma$ is the softplus function. Finally, $G$ and the basis function number are concatenated and sent to a full-connected layer to get the predictive time.
D. *Adaption of ML/AI models for the chemical MWI ansatz*

The ML/AI models can be expected to supply reliable predictions for molecules in the world of the same basis set, same functional, same convergence strategy, etc. when training the models. However, there are various basis sets, functionals etc., and as such, their combinations should be innumerable. It means the forecasting system should have the potential to supply reliable predictions for molecules even without pre-trained models.

In the chemical MWI ansatz, all the worlds share a same and unique start-point, which includes the molecules to be calculated and the KS equations of Eq.(3). Considering all the result’s worlds stem from the same molecules and KS equations, then for the worlds contain the similar parameters should "close" to each other. For example, assuming there are two result’s worlds with identical computational parameters but different basis sets of cc-pVDZ\(^{48}\) and def2-SVP\(^{49}\), then for organic systems one can expect similar computational times. Because the number of functions of cc-pVDZ basis sets are 5 for H-He (2s1p atomic basis functions contracted from 4s1p primitive Gaussian functions, i.e. (4s1p) \(\rightarrow\) [2s1p]), 14 for Li-Ne ((9s4p1d) \(\rightarrow\) [3s2p1d]) and these of def2-SVP basis sets are 5 for H-He ((4s1p) \(\rightarrow\) [2s1p]), 14 for Li-Ne ((7s4p1d) \(\rightarrow\) [3s2p1d]) — these two cases own similar basis sets structure thus a similar time scaling can be expected. The same approximation way can be used for the molecules with only different DFT functionals, and we treat this way as the *similarity* strategy as shown in the left part of FIG.7.

Beyond the above mention *similarity* way, we also try to discover the connections between different worlds via the typical molecules as the joint points. As shown in the right part of FIG.7, series of super-surfaces can absorb the identical molecules from different worlds, within each super-surface one can make the relationship among different worlds. For example, one can roughly fit between number of basis and computational time, and we treat this way as the "fitting" strategy. Currently, the *fitting* strategy use the fitted regress equation of \(C_{10}H_{11}NOS\) in its super-surface (i.e. \(y = 5.95x - 644\) with time \(y\) and number of basis functions \(x\)) according to the number of basis sets. For DFT functionals, currently we simple use the Jacob ladder as the classifier. For example, any GGA functionals share the same scale parameters and different to these of LDA functionals.
FIG. 7. The illustration of fitting and similarity strategies in the chemical MWI ansatz.

FIG. 8. The fitting basis sets and the fitted regress equation used in the fitting strategy in this work.

E. Workflow of the established forecasting System

The workflow of our forecasting system and the related supporting packages (also explained in the Sec.III) are illustrated in FIG. 9. For any input molecule, the forecasting system can supply a predicted computational time under the desired DFT-level.

III. COMPUTATIONAL DETAILS

We use the BASIS SET EXCHANGE\textsuperscript{50}, a community database for quantum chemistry electronic structure calculations to obtain the information of basis sets and electrons. The STK\textsuperscript{51} package together with the RDKIT\textsuperscript{52} package are used for generating the molecular suites. These two packages are also used for extracting and labeling properties for the molecular suites. All the calculations are implemented by the GAUSSIAN.09 package. The self-written scripts using PYTHON with NUMPY, TENSORFLOW\textsuperscript{53} and PYTORCH\textsuperscript{54} are used for automatically implementing the calculations, assembling the data, as well as
analysing the results.

The training sets of RF with FNN model is self-designed, containing 139 molecules with typical structures, and of Bi-LSTM, MPNN, and MGCN models contains 1639 molecules sampled from alchemy dataset\textsuperscript{55}, respectively. The test sets contains 317 molecules which are also from alchemy dataset, these test sets are used for all the models. When training the models, the width of the hidden layers was set to 5 (means that every hidden layer has 5 neurons). The mean squared error (MSE) loss was used as the target function, and the gradient descent algorithm was used for minimizing the loss. The learning rate of models (i.e. cost function) was set to 0.01, and the number of epochs was 2000, respectively. The coefficients for each "cost basis" are obtained via the RF classifier with the \textsc{scikit-learn}(\textsc{sklearn})\textsuperscript{56} package, with all the parameters set to default values. The classifier provides a molecule’s probabilities of falling into each category as the output, which can be used in Eq. (8) for the predictions. During the training of the Bi-LSTM model, the dimension of LSTM layers was set to 100 and the number of full connected layers is set to 5. The previous 4 full connected layers had the width of 10 neurons and the last has the width of 5 neurons. The loss function was MSE with the Adam algorithm as the optimizer. The learning rate was 0.01, the batch size was 128, and number of epochs was 25. Finetuning was also done afterwards, with the last two full connected layers trained and all the previous layers’ parameters (including LSTM layers,
the attention layer and other full connected layers) fixed. The learning rate was 0.001 and the number of epochs was 30 at the finetuning step.

In the process of training the MPNN model, which was modified from original code of Tencent quantum laboratory, the dimension of hidden states of nodes was set to 128. The number of message passing steps were set to 6. The number of set2set steps was 6, too. A 64 neurons wide full connected layer was constructed before the final output layer. All the activate functions were set to RELU function. The loss function was MSE loss. The optimizer was Adam with the learning rate of 0.001. The number of epochs was 7. At the finetuning step, the full connected layer was trained at the learning rate of 0.001 in 6 epochs.

When training the MGCN model, which was also modified from original code of Tencent quantum laboratory, both the dimensions of the atom embedding and the edge embedding were set to 128. The interaction layers were set to 3 levels. The loss function was MSE loss. The optimizer was Adam, with the learning rate of 0.001. The number of epochs was 50. At the finetuning step, the full connected layer was trained with the learning rate of 0.001 in 35 epochs.

IV. RESULTS AND DISCUSSIONS

A. Predicting with pre-trained models

In TABLE I, the mean relative errors (MRE) and variances (Δ) of test sets are listed. On average, The RF with FNNs model has the best performance when comparing to Bi-LSTM, MPNN, and MGCN models. The Bi-LSTM, MPNN, and MGCN perform a little more unsteadily across the basis sets or functional types, such as the Bi-LSTM, MPNN show MREs larger than 30% in B3LYP/6-31g* case, and the same for the MGCN model in the M06-2x/6-31g case. However, these separate fluctuations should be acceptable for predicting purpose. It is noteworthy that most of the molecules in our datasets are drug molecules, although RF classifier is not specialized for drug molecules, it can still has the capacity for analyzing the structural features and supply reasonable predicted values. Additionally, the average MREs are about 16% for RF case and about 20% for Bi-LSTM, MPNN, and MGCN cases. These values should be acceptable in our cognition for the
prediction purpose.

TABLE I. The MRE and variances (Δ) for the ML/AL models that used in predicting the computational time for various basis sets and functionals

| Functional | Basis set | RF MRE |  | RF ΔMRE |  | Bi-LSTM MRE |  | Bi-LSTM ΔMRE |  | MPNN MRE |  | MPNN ΔMRE |  | MGCN MRE |  | MGCN ΔMRE |
|------------|-----------|--------|---|--------|---|-------------|---|-------------|---|---------|---|---------|---|---------|---|
| CAM-B3LYP  | 6-31g*    | 0.1686 | 0.0152 | 0.1580 | 0.0116 | 0.2102 | 0.0199 | 0.2602 | 0.0405 |
| B3LYP      | 6-31g*    | 0.1457 | 0.0157 | 0.4056 | 0.0234 | 0.3252 | 0.0286 | 0.1550 | 0.0150 |
| M06        | 6-31g*    | 0.2055 | 0.0224 | 0.1717 | 0.0107 | 0.1983 | 0.0202 | 0.2111 | 0.1011 |
| M06-2x     | 6-31g     | 0.1825 | 0.0190 | 0.1988 | 0.0165 | 0.2037 | 0.0183 | 0.3708 | 0.0301 |
| M06-2x     | 6-31g*    | 0.1971 | 0.0232 | 0.1868 | 0.0113 | 0.2096 | 0.0172 | 0.2418 | 0.0705 |
| M06-2x     | 6-31+g*   | 0.1295 | 0.0124 | 0.1347 | 0.0050 | 0.1348 | 0.0144 | 0.1785 | 0.1290 |
| M06-2x     | 6-31g**   | 0.1279 | 0.0118 | 0.1960 | 0.0133 | 0.1946 | 0.0151 | 0.1366 | 0.0416 |
| Average    |           | 0.1653 | 0.0171 | 0.2075 | 0.0131 | 0.2105 | 0.0191 | 0.2219 | 0.0611 |

In FIG[10] we show the normal distributions of MRE under the three models with the combination of M06-2X/6-31g*. The most frequent values fall into the range of -0.1∼0, 0.1∼0.2 and 0.2∼0.3 for the four models, respectively. The range of relative errors from RF model is narrower than these from other models, although the errors values in RF model tend to be negative. Errors of Bi-LSTM model are more deviated to the positive direction. The MPNN model has its errors uniformly distributed across the domain of values from -0.3 to 0.4, which corresponds to its higher variance but it is consistent with normal distribution. The other graph-based MGCN model even worse than that of MPNN for both averaged MRE and variance, it may imply that current fuzzy parameters like total number of basis sets can not functionalize the MPNN and MGCN models. Converting the basis sets information into graphs should be a potentional cure.

Because all the results in TABLE[II] were obtained in the same HPC cluster with 2-way E5-2609v3 processors and 16GB memory, we also used different HPC clusters (2-way E7-4830v4 and 2-way Sliver-4116 clusters) in order to check the robustness. The results are listed in TABLE[II] in which all the calculations were implemented using M06-2x functional and 6-31g* basis sets via 12 threads. It can be found that although there are fluctuations (maybe caused by the turbo mode or workloads) in the MRE values, they are
FIG. 10. The illustration of normal distributions of relative errors for these ML/AI models still range from 13% ∼ 34% and satisfy the purpose for predicting.

TABLE II. The MRE and variances (Δ) for the ML/AL models that used in predicting the computational time for different HPC clusters

| HPC cluster  | RF       | Bi-LSTM  | MPNN    | MGCN    |
|--------------|----------|----------|---------|---------|
|              | MRE      | Δ        | MRE     | Δ        | MRE     | Δ        |
| E5-2609v3    | 0.1971   | 0.0232   | 0.1868  | 0.0113   | 0.2096  | 0.0172   | 0.2418  | 0.0705   |
| E7-4830v4    | 0.2126   | 0.0166   | 0.2077  | 0.0212   | 0.2405  | 0.0299   | 0.1890  | 0.0280   |
| Sliver-4116  | 0.1328   | 0.0133   | 0.3404  | 0.0250   | 0.1445  | 0.0146   | 0.1378  | 0.0331   |
| Average      | 0.1808   | 0.0177   | 0.2450  | 0.0192   | 0.1973  | 0.0206   | 0.1895  | 0.0439   |

B. Predicting without pre-trained models

All the above-mentioned ML/AI models were trained in a given world in the chemical MWI ansatz, which means the basis set, functional, etc. were fixed and same between training sets and test sets. In order to make predictions using the untrained computa-
tional parameters, we employed the "similarity" and "fitting" strategies as introduced in Sec.II.D, and the results are listed in TABLE III.

For the similarity case, the trained models of adjacent world were used. For example, in the M06-2x/cc-pVDZ case, the model of M06-2x/def2-SVP case was used as the alternative model for predicting, and vice versa for the M06-2x/def2-SVP case (it means the model of M06-2x/cc-pVDZ case was used). Since the basis sets of cc-pVDZ and def2-SVP share quite similar constitutions in primitive Gaussian functions together with same constitutions in contracted atomic basis functions, one can find that all prediction ways, whatever in the original Bi-LSTM case, similarity case, or fitting case, can have quite good predictions with MRE around 0.20.

In the fitting case, the predictions were obtained by the linear combinations of models from nearby worlds using the fitting weights. It implies that this way should owns larger flexibility if unknown basis sets were used, although the precision may loss a bit. This deduction can be verified when looking the TABLE III. For example, in the M06-2x/cc-pVDZ case, the fitting strategy give a MRE of 0.1759 and Δ of 0.0154, which is a bit larger than these in the similarity case. For others results, similar phenomenon can be observed.

TABLE III. The MRE and variances (Δ) for the test sets between Bi-LSTM model and "similarity" and "fitting" strategies, which can be used to predict the computational time without pre-trained models in the aiming world.

| Functional Basis set | Bi-LSTM | Similarity strategy | Fitting strategy |
|----------------------|---------|---------------------|-----------------|
|                      | MRE  | Δ   | MRE  | Δ   | MRE  | Δ   |
| M06-2x cc-pVDZ       | 0.0834 | 0.0039 | 0.1355 | 0.0071 | 0.1759 | 0.0154 |
| M06-2x def2-SVP      | 0.0828 | 0.0038 | 0.2356 | 0.0091 | 0.1198 | 0.0123 |
| wb97xd 6-31G*        | 0.2277 | 0.0127 | -     | -     | 0.3751 | 0.0190 |
| wb97xd def2-SVP      | 0.1689 | 0.0099 | 0.1171 | 0.0069 | 0.1454 | 0.0158 |

V. CONCLUSION

Inspired by the MWI of the multiverse, we proposed the chemical MWI ansatz, within which the ML/AI methods and cheminformatics were employed, in order to develop a
forecasting systems for predicting DFT computational times in quantum chemical studies. In the chemical MWI model, we assumed the molecules and KS equation forms the trunk and various parameters in solving the KS equations generate various branches or “worlds” that contain all the solved wave function and the corresponding computational times. Within given worlds, several ML/AI models can be trained in order to give reliable predictions of the computational times. These trained models in given worlds can also be used to give predictions for the molecules in the worlds without pre-trained models, so that this forecasting systems have the potential to give predictions for any computational parameters in the KS frameworks.

Four ML/AI models, including RF, Bi-LSTM, MPNN, and MGCN models, were used as kernels for running the forecasting system. The typical MREs were 20%–30% for these models, and the RF model showed the best capability among these models with MRE around 15% and have the potential to be employed in various molecules systems.

Nevertheless, we should mention that the training sets were from alchemy dataset of Tencent for Bi-LSTM, MPNN and MGCN models, as such, the transferability of predictions in different molecules systems should be limited, unless massive amounts of training sets involved. Beyond that, more elaborate operations should be considered under the chemical MWI ansatz in our following work, especially how to take account the parameters of primitive Gaussian functions and the contracted atomic functions into the forecasting systems.

VI. ACKNOWLEDGEMENT

The work was supported by the National Key Research and Development Program of China (Grant NO.2018YFB0203805), National Natural Science Foundation of China (Grant NO.21703260), and the Informationization Program of the Chinese Academy of Science (Grant NO.XXH13506-403). We also thank PARATERA company for the cooperation.
REFERENCES

1. Friesner, R. A. Ab initio quantum chemistry: Methodology and applications. *Proc. Natl. Acad. Sci. U.S.A.* **2005**, *102*, 6648–6653.

2. Hergert, H.; Bogner, S.; Morris, T.; Schwenk, A.; Tsukiyama, K. The In-Medium Similarity Renormalization Group: A novel ab initio method for nuclei. *Phys. Rep.* **2016**, *621*, 165 – 222, Memorial Volume in Honor of Gerald E. Brown.

3. Al-Douri, Y.; Hashim, U.; Khenata, R.; Reshak, A.; Ameri, M.; Bouhemadou, A.; Ruslinda, A. R.; Arshad, M. M. Ab initio method of optical investigations of CdS1-xTex alloys under quantum dots diameter effect. *Sol. Energy* **2015**, *115*, 33 – 39.

4. Scerri, E. R. Has chemistry been at least approximately reduced to quantum mechanics? PSA: Proceedings of the biennial meeting of the philosophy of science association. 1994; pp 160–170.

5. Bash, P. A.; Ho, L. L.; MacKerell, A. D.; Levine, D.; Hallstrom, P. Progress toward chemical accuracy in the computer simulation of condensed phase reactions. *Proc. Natl. Acad. Sci. U.S.A.* **1996**, *93*, 3698–3703.

6. Zhao, L.; Pan, S.; Holzmann, N.; Schwerdtfeger, P.; Frenking, G. Chemical bonding and bonding models of main-group compounds. *Chem. Rev.* **2019**, *119*, 8781–8845.

7. Moore, G. E., et al. Cramming more components onto integrated circuits.

8. Moore, G. E., et al. Progress in digital integrated electronics. Electron Devices Meeting. 1975; pp 11–13.

9. Strout, D. L.; Scuseria, G. E. A quantitative study of the scaling properties of the Hartree–Fock method. *J. Chem. Phys.* **1995**, *102*, 8448–8452.

10. Li, W.; Duan, M.; Liao, K.; Hong, B.; Ni, Z.; Ma, J.; Li, S. Improved generalized energy-based fragmentation approach and its applications to the binding energies of supramolecular complexes. *Electron. Struct.* **2019**, *1*, 044003.

11. Godbout, N.; Salahub, D. R.; Andzelm, J.; Wimmer, E. Optimization of Gaussian-type basis sets for local spin density functional calculations. Part I. Boron through neon, optimization technique and validation. *Can. J. Chem.* **1992**, *70*, 560–571.

12. Li, W.; Chen, C.; Zhao, D.; Li, S. LSQC: Low scaling quantum chemistry program. *Int. J. Quantum Chem.* **2015**, *115*, 641–646.
13 Møller, C.; Plesset, M. S. Note on an Approximation Treatment for Many-Electron Systems. *Phys. Rev.* **1934**, *46*, 618–622.

14 Purvis, G. D.; Bartlett, R. J. A full coupled-cluster singles and doubles model: The inclusion of disconnected triples. *J. Chem. Phys.* **1982**, *76*, 1910–1918.

15 Cullen, J. M.; Zerner, M. C. The linked singles and doubles model: An approximate theory of electron correlation based on the coupled-cluster ansatz. *J. Chem. Phys.* **1982**, *77*, 4088–4109.

16 Raghavachari, K.; Trucks, G. W.; Pople, J. A.; Head-Gordon, M. Reprint of: A fifth-order perturbation comparison of electron correlation theories. *Chem. Phys. Lett.* **2013**, *589*, 37–40.

17 Harding, M. *Large-scale coupled-cluster calculations*; Univ.: Mainz, 2008; p 5.

18 Xiao, H.; Wu, H.; Chi, X. SCE: grid environment for scientific computing. International Conference on Networks for Grid Applications. 2008; pp 35–42.

19 Raicu, I.; Foster, I. T.; Zhao, Y. Many-task computing for grids and supercomputers. 2008 workshop on many-task computing on grids and supercomputers. 2008; pp 1–11.

20 Yang, X.-J.; Liao, X.-K.; Lu, K.; Hu, Q.-F.; Song, J.-Q.; Su, J.-S. The TianHe-1A supercomputer: its hardware and software. *J COMPUT SCI TECH-CH* **2011**, *26*, 344–351.

21 Rodrigues, E. R.; Cunha, R. L. F.; Netto, M. A. S.; Spriggs, M. Helping HPC Users Specify Job Memory Requirements via Machine Learning. 2016 Third International Workshop on HPC User Support Tools (HUST). 2016; pp 6–13.

22 Witt, C.; Bux, M.; Gusew, W.; Leser, U. Predictive performance modeling for distributed batch processing using black box monitoring and machine learning. *Inf Syst* **2019**, *82*, 33–52.

23 Nemirovsky, D.; Arkose, T.; Markovic, N.; Nemirovsky, M.; Unsal, O.; Cristal, A.; Valero, M. A general guide to applying machine learning to computer architecture. *Supercomput Front Innov* **2018**, *5*, 95–115.

24 Garg, S. K.; Yeo, C. S.; Anandasivam, A.; Buyya, R. Environment-conscious scheduling of HPC applications on distributed Cloud-oriented data centers. *J Parallel Distrib Comput* **2011**, *71*, 732–749.

25 Papay, J.; Atherton, T. J.; Zemerly, M. J.; Nudd, G. R. Performance prediction of parallel self consistent field computation. *Parallel Algorithms Appl.* **1996**, *10*, 127–143.
Antony, J.; Rendell, A. P.; Rui, Y.; Trucks, G.; Frisch, M. J. Modelling the Runtime of the Gaussian Computational Chemistry Application and Assessing the Impacts of Microarchitectural Variations. *Procedia Comput. Sci.* 2011, 4, 281–291.

Mniszewski, S. M.; Junghans, C.; Voter, A. F.; Perez, D.; Eidenbenz, S. J. TADSim: Discrete Event-Based Performance Prediction for Temperature-Accelerated Dynamics. *ACM Trans Model Comput Simul* 2015, 25, 1–26.

Heinen, S.; Schwilk, M.; von Rudorff, G. F.; von Lilienfeld, O. A. Machine learning the computational cost of quantum chemistry. *arXiv preprint arXiv:1908.06714* 2019,

Vaidman, L. In *The Stanford Encyclopedia of Philosophy*, fall 2018 ed.; Zalta, E. N., Ed.; Metaphysics Research Lab, Stanford University, 2018.

Laszlo, E. *Connectivity Hypothesis, The: Foundations of an Integral Science of Quantum, Cosmos, Life, and Consciousness*; SUNY Press, 2003; p 108.

Steele, J. M. *The Cauchy-Schwarz master class: an introduction to the art of mathematical inequalities*; Cambridge University Press, 2004.

Everett, H. “Relative State” Formulation of Quantum Mechanics. *Rev. Mod. Phys.* 1957, 29, 454–462.

Varadharajan, A.; Rajesh, R. Chemoinformatics — A quick review. *ICECT 2011 - 2011 3rd International Conference on Electronics Computer Technology* 2011, 6.

Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Model.* 1988, 28, 31–36.

Weininger, D.; Weininger, A.; Weininger, J. L. SMILES. 2. Algorithm for generation of unique SMILES notation. *J. Chem. Inf. Model.* 1989, 29, 97–101.

Weininger, D. SMILES. 3. DEPICT. Graphical depiction of chemical structures. *J. Chem. Inf. Model.* 1990, 30, 237–243.

Dalby, A.; Nourse, J. G.; Hounshell, W. D.; Gushurst, A. K.; Grier, D. L.; Leland, B. A.; Laufer, J. Description of several chemical structure file formats used by computer programs developed at Molecular Design Limited. *J. Chem. Inf. Model.* 1992, 32, 244–255.

Tin Kam Ho, Random decision forests. Proceedings of 3rd International Conference on Document Analysis and Recognition. 1995; pp 278–282 vol.1.

Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* 1997, 9, 1735–1780.
40. Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural Message Passing for Quantum Chemistry. *CoRR* 2017, *abs/1704.01212*.

41. Lu, C.; Liu, Q.; Wang, C.; Huang, Z.; Lin, P.; He, L. Molecular Property Prediction: A Multilevel Quantum Interactions Modeling Perspective. *Proceedings of the AAAI Conference on Artificial Intelligence* 2019, 33, 1052–1060.

42. Zhou, P.; Shi, W.; Tian, J.; Qi, Z.; Li, B.; Hao, H.; Xu, B. Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Berlin, Germany, 2016; pp 207–212.

43. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781 2013*.

44. Cho, K.; Van Merriënoor, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259 2014*.

45. Vinyals, O.; Bengio, S.; Kudlur, M. Order Matters: Sequence to sequence for sets. *CoRR* 2015, *abs/1511.06391*.

46. Broomhead, D. S.; Lowe, D. *Radial basis functions, multi-variable functional interpolation and adaptive networks*; 1988.

47. Schwenker, F.; Kestler, H. A.; Palm, G. Three learning phases for radial-basis-function networks. *Neural Netw.* 2001, 14, 439–458.

48. Dunning, T. H. Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen. *J. Chem. Phys.* 1989, 90, 1007–1023.

49. Zheng, J.; Xu, X.; Truhlar, D. Minimally augmented Karlsruhe basis sets. *Theor. Chem. Acc.* 2011, 128, 295–305.

50. Pritchard, B. P.; Altarawy, D.; Didier, B. T.; Gibson, T. D.; Windus, T. L. A New Basis Set Exchange: An Open, Up-to-date Resource for the Molecular Sciences Community. *J. Chem. Inf. Model.* 2019,

51. Turcani, L.; Berardo, E.; Jelfs, K. STK: A Python Toolkit for Supramolecular Assembly. 2018,

52. The RDKit Documentation — The RDKit 2019.03.1 documentation. [http://www.rdkit.org/docs/index.html](http://www.rdkit.org/docs/index.html).

53. Abadi, M. et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015; [https://www.tensorflow.org/](https://www.tensorflow.org/), Software available from tensorflow.org.
54Paske, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. 2017,

55Chen, G.; Chen, P.; Hsieh, C.-Y.; Lee, C.-K.; Liao, B.; Liao, R.; Liu, W.; Qiu, J.; Sun, Q.; Tang, J.; Zemel, R.; Zhang, S. Alchemy: A Quantum Chemistry Dataset for Benchmarking AI Models. arXiv preprint arXiv:1906.09427 2019,

56Pedregosa, F. et al. Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. 2011, 12, 2825–2830.

57Tencent Quantum laboratory. https://alchemy.tencent.com/