DFTB simulation of charged clusters using machine learning charge inference

Paul Guibourg, Léo Dontot, Pierre-Matthieu Anglade, Benoit Gervais

DFTB simulation of charged clusters using machine learning charge inference

Paul Guibourg, Léo Dontot, Pierre-Matthieu Anglade,* and Benoit Gervais

Laboratoire Cimap, UMR6252 — Université de Normandie Caen, École supérieure d'ingénieures de Caen, Commissariat à l'énergie atomique, Centre national de la recherche scientifique — 6 Boulevard du Maréchal Juin, 14050 Caen Cedex, France

E-mail: Pierre-Matthieu.Anglade@unicaen.fr

Phone: +33 (0)2 31452665

Abstract

We present a modification to Self-Consistent Charge Density Functional based Tight Binding (SCC-DFTB), which allows computation based on approximate atomic charges. We obtain these charges by means of a machine learning (ML) process, which combines a Coulomb model with a neural network. This allows us to avoid the Self-Consistent Charge (SCC) cycles in SCC-DFTB calculation, while keeping its accuracy. The main input of the model are the atomic positions characterized by a set of Atom-Centred Symmetry Functions (ACSF). The charge inference from our ML algorithm is as close as 10^{-2} unit of charge from the exact SCC solution. Our ML-DFTB approach provides a good approximation of the density matrix and of the energy and forces with only one single diagonalization. This is a significant computational saving with respect to the complete SCC algorithm, which allows us to investigate bigger ensemble of atoms. We show the quality of our approach in the case of charged Silicon Carbide (SiC) clusters. The ML-DFTB Potential Energy Surface (PES) mimics rather well the SCC-DFTB PES despite its simplicity. This allows us to obtain the same geometric structure

ordering with respect to energy for small clusters. The dissociation barriers for ion emission are well reproduced, which opens the way to investigate ion field emission and charged cluster stability. The ML-DFTB approach is obviously not limited to charged clusters nor to SiC materials. It opens a new route to investigate larger clusters than investigated by standard SCC-DFTB, as well as surface and solid state chemistry at the atomic level.

1 Introduction

Machine learning algorithms have recently appeared in the field of quantum chemistry and they offer a possible route to increase the size of molecules and atomic clusters investigated by means of quantum chemical calculations. An obvious use of machine learning is perhaps finding the functional for DFT calculation. Such difficult research is still in its infancy[1]. Following a different route, several authors use it to generate new kind of Machine Learning Potentials[2,3] (MLP). There are two kinds of neural network. The first series uses static descriptors while the second series uses self descriptors. The latter are usually based on a Message Passing Neural Network (MPNN) architecture. The first neural network proposed for atomic structure determination was the Deep Tensor Neural Network proposed in 2017 by Schütt *et al.*[4]. It was soon followed by several ML approaches like SchNet[5], PhysNet[6] and AIMNet[7]. These models are used to provide ML inference of energy and forces, and sometimes atomic charges. ML is not restricted to NN. Alternative regression models, like gaussian regression potential can also be used to compute energy, forces, and other quantities like the electronic density[8]. Note however that such kind of algorithms do not predict other properties related to the electronic density, which are by construction not available. Moreover, the total charge of a cluster, when predicted, is usually not conserved in these models and an alternative approach is desirable to study charged molecules and clusters with a prescribed charge.

To build a charge-conserving algorithm, we choose to adapt the Charge Equilibration via

NN Technique[9] (CENT) model based on High Dimension Neural Network[10] (HDNN) often used to predict potential energy surfaces. In particular, the fourth generation of HDNN models infer the energy from atomic charges like in CENT[11]. The atomic charges are thus used in this model to compute the energy, with an accuracy of the order of 2 meV as discussed by Faber *et al.*[12]. The link with electronic structure methods based on explicit atomic charges is rather natural. In particular, the link with Density Functional based on Tight-Binding[13–15] (DFTB) model, as this method is based on atomic charge definition to introduce flexibility in the Tight-Binding (TB) method. A good ML inference of atomic charges could thus be followed by a DFTB calculation based on the potential generated by these charges, as in the full Self-Consistent Charge (SCC-)DFTB approach. Moreover, in SCC-DFTB, the computational bottleneck is in the self-consistent determination of the charges, which requires iterative computation of a potentially large eigenvalue problem for large atomic system, despite the reduced basis set. It is thus interesting to bypass as much as possible the SCC iterations to get the electronic density. This is the route we shall follow here, by combining an atomic charge prediction algorithm with SCC-DFTB.

Previous attempts to use machine learning in the framework of DFTB focus mainly on the improvement the DFTB accuracy with respect to DFT reference results. Some authors choose to modify the original Hamiltonian and overlap matrix element to fit better the energy and dipole of small organic molecules[16]. Some others follow a similar route to reproduce better the density of states, in particular for SiC material[17]. Some other authors follow a simpler alternative route, which consists in obtaining an improved repulsive function with the aim of reducing the difference between DFT and DFTB energies for various crystallographic phase of pure silicon[18].

In the present work we do not attempt to improve the DFTB parametrization. We limit ourselves to derive a model as close as possible to the targeted SCC-DFTB reference while using a charge inference; our work should be considered a proof of concept.

In this paper, we focus on charged clusters. The interest in small charged objects has

experienced a complete renewal at the end of the twentieth century with the discovery of new methods to produce atomic and molecular clusters[19]. The recent progresses in this field allow chemists to produce an incredibly large variety of clusters covering a very broad range of size from a few-atom clusters to dust grains[20,21]. Regarding the chemical nature of these clusters, almost all kind of chemical bonding can be found, from weekly bound van der Waals to tightly bound ionic clusters, and from covalent to metallic clusters[20–23]. Upon charging, either suddenly or by means of a soft nearly adiabatic process, a cluster is going to evolves to release the excess energy accumulated by charging. It will thus deform and eventually release the charge excess by ion emission. Such a charge emission process is a quite general physical process, which can be observed as soon as a material surface experiences a sufficiently strong electric field. This fundamental process controls the charge over mass limit sustainable by an assembly of atoms, which considerably depends on the chemical nature of the inter-atomic forces[19]. In the field of material science, the charge instability finds a metrology application with its use in Atom Probe Tomography[24] (APT). In this case, the application of a large electric potential to a very sharp tip allows scientists to perform three-dimensional chemical analysis of materials at the nanometre scale. The stability of weakly charged molecular clusters also presents a strong interest in mass spectrometry as it is currently used to identify molecular objects[21].

Theoretical investigation of charged clusters was so far relatively limited. For few-atoms clusters the method of quantum chemistry based either on wave function or Density Functional Theory (DFT) are often used. Such a limited number of atoms barely represent a surface of a sample with terraces or defects due to the erosion induced by atom emission and it is difficult to achieve a comparison with experiment obtained with such kind of samples. For large atomic systems, theoretical modelling often resort to *ad hoc* force fields, which are limited in their transferability to one single class of materials for a given force field. To overcome this limitation, we use the SCC-DFTB to simulate the electronic structure of clusters made of few hundred atoms. It avoids the bottleneck of integral calculation, which limits *ab*

initio methods. It also limits the rank of the algebraic problem by using a minimal basis set and restores to some extent the accuracy by using *ad hoc* short range repulsion terms. It provides thus a minimal description of chemical bonding at a cheap computational cost and we can use it to generate the potential energy surface of such a system as well as the forces experienced by each atom, like in *ab initio* molecular dynamics. Moreover, the DFTB material parameters data are rather rich[25], and the method offers thus the possibility to study many materials of different nature, which allows a sensible comparison with experiments.

Our machine learning algorithm combines atomic environment data based on Atom-Centered Symmetry Functions[26] (ACSF) and a simple Coulomb model described in section 2. The ML algorithm presented in section 3 provides us a computationally cheap first guess of the atomic charges, which are used as input for a SCC-DFTB computation of the electronic density, and then the energy and the atomic forces. The quality of the charge prediction algorithm with respect to the SCC algorithm convergence is discussed in section 3. As an example, we investigated Silicon Carbide (SiC) charged clusters in section 4.

This semiconducting material is an interesting test case for APT where the analysis reveals some composition bias which might be due to field induced atomic reorganization at the surface[27]. Moreover, we use the existing parameterization for SCC-DFTB[25]. The quality of the simulation is discussed for a few examples of cluster geometries obtained with ML-DFTB and compared with SCC-DFTB and DFT results. We also demonstrate in section 5 the fidelity of the simulation to reproduce dissociation energy barriers, which are of prime interest for APT. Finally, our general conclusions and perspectives about the ML-DFTB method are given in section 6.

2 Model

2.1 Atomic charge inference

Our model is a two-step model, which combines a machine learning inference of atomic charges and the determination of the energy with the help of DFTB computation using the inferred charges. The first step is similar to the CENT algorithm[9] adapted in our case to the prediction of atomic charges for a cluster of total charge Q_{tot}. The atomic charges $\mathbf{q} = \{q_A\}$ are obtained by minimization of the Lagrangian $\mathcal{L}$ defined as:

$$\mathcal{L} = \boldsymbol{\chi}\mathbf{q} + \frac{1}{2}\mathbf{q}\gamma\mathbf{q} + \lambda\left(\mathbf{q}\mathbf{r} - Q_{tot}\right) \tag{1}$$

The CENT model[9] has proven to be quite useful for ionic clusters[28] and crystals[29]. It is a ML adaptation of charge equilibration method[30,31], which has been used to predict constrained atomic charges and to deduce the energy of the system. In this method, the quadratic term can be regarded as the Coulomb interaction between atomic sites with a site occupation controlled by a Hubbard parameter and the linear term can be considered as an approximate band energy linearized with respect to atomic charge variation. There is however a major difference between our approach and the CENT method, as we do not attempt to get the energy from the quadratic charge dependence implied by equation 1. On the contrary, we use the inferred charges as input for a DFTB calculation.

In the above equation, the set of linear term $\boldsymbol{\chi} = \{\chi_A\}$ is provided by machine learning. Note that $\boldsymbol{\chi}$ is merely a fitting parameter which does not represent a true physical quantity, like the DFTB band energy. The second term is simply the SCC energy as defined in the SCC-DFTB method[13]. The matrix γ reduces to the pure Coulomb interaction of unit charges at large distance and is constrained to the Hubbard parameter for on-site diagonal elements. It takes into account the long range Coulomb interaction, which is essential for charged clusters. The last term in equation 1 is simply the charge constraint with the Lagrange multiplier λ and the vector $\mathbf{r} = (1, 1, 1, ..., 1)$ stands for the N-rank vector filled with 1

so that $\mathbf{qr} = \sum_A q_A$. From a numerical point of view, the simple quadratic form of the Lagrangian is particularly convenient to find the solution $\mathbf{q}$ that minimizes it. Moreover, we shall see that despite its simplicity, it produces a rather good approximation of atomic charges q_A to be used in SCC-DFTB calculation.

We can easily eliminate λ by means of a few algebraic manipulations. We can show that:

$$\lambda = -\frac{Q_{tot} + \mathbf{r}\gamma\boldsymbol{\chi}}{\mathbf{r}\gamma\mathbf{r}} \tag{2}$$

$$\mathbf{q} = -\gamma(\boldsymbol{\chi} + \lambda\mathbf{r}) \tag{3}$$

Hence, the charge vector $\mathbf{q}$ is completely defined by the vector $\boldsymbol{\chi}$. We obtain the necessary values of $\boldsymbol{\chi}$ from the machine learning algorithm defined in section 3. The input of the algorithm are the positions of the atoms characterized by the ACSF [26] and the chemical nature of the atoms, either C or Si in the present case. The output of the ML algorithm is a complete set of parameter χ_A for each atom A. Thus, for each atom A we obtain a charge q_A, which depends on the environment of the atom A. A comparison of the charge distribution with reference charges is shown in figure 1 for SiC clusters for our training set made of $8 \cdot 10^4$ clusters. The typical dispersion is of the order ± 0.01 unit of charge, for an average charge per atom of the order of 0.10. We have also plotted, in figure 2, the energy and forces of the ML-DFTB with respect to SCC-DFTB. Despite a small systematic shift of the energy, the global agreement is better than the charge agreement.

Once we have obtained the atomic charges $\{q_A\}$, we use them as input for a SCC-DFTB calculation with an improved guess. This guess of charge should ideally match the exact charge, which result from the SCC-DFTB calculation, so that the iterative self-consistent energy minimization process would reduce to one single step. There are however unavoidable fluctuations inherent to the machine learning process. We have thus two alternatives. We can either use the machine learning charges to perform a single diagonalization and determine

approximate energy and forces as detailed in section 2.2, or perform a complete SCC cycle with the benefits of an improved charge guess as detailed in section 2.3.

2.2 Bypassing SCC calculation

Once obtained from ML, the atomic charges $\mathbf{q}$ are used in a parametric Hamiltonian $h(\mathbf{q})$. We then perform only one diagonalization of this Hamiltonian to compute the DFTB orbital coefficients $c_{n\mu}$. In this way, our model reduces to a standard tight-binding calculation regarding the computational cost. The Hamiltonian $h(\mathbf{q})$ is defined as in SCC-DFTB:

$$h_{\mu\nu}(\mathbf{q}) = h_{\mu\nu}(\mathbf{0}) + \frac{1}{2}\sum_{AB}\left(q_A\gamma_{AB}S_{B,\mu\nu} + S_{A,\mu\nu}\gamma_{AB}q_B\right) \tag{4}$$

where $h(\mathbf{0})$ is the band Hamiltonian and the second term comes from the derivative of the SCC term with respect to the orbital coefficients, with $S_{A,\mu\nu}$ the overlap matrix related to the basis functions centered on atom A and conversely for $S_{B,\mu\nu}$ with respect to atom B. The choice of this matrix corresponds to Mulliken definition of atomic charge, which reads:

$$S_{A,\mu\nu} = \frac{1}{2}\left(S_{\mu\in A\nu} + S_{\mu\nu\in A}\right) \tag{5}$$

Denoting $\rho_{\mu,\nu}$ a density matrix element, the total SCC-DFTB energy reads:

$$E = \sum_{\mu\nu}\rho_{\mu\nu}h_{\mu\nu}(\mathbf{0}) + \frac{1}{2}\sum_{AB}q_A\gamma_{AB}q_B \tag{6}$$

A straightforward use of eq. 6 as the energy definition, by simply substituting the ML charges in the Coulomb term, is not consistent with the SCC-DFTB. In such a case, the Coulomb term no longer depend on the density matrix and implicitly on the related orbital coefficients c^n_μ. Thus, the minimization with respect to the orbital coefficients c^n_μ would produce a set of molecular orbitals $\{\phi_n\}$ solution of $h(0)\phi_n = \epsilon_n\phi_n$, which are quite different from the true SCC-DFTB molecular orbitals produced using $h(q)$ defined by eq. 4. Hence,

the energy as defined by eq. 6 would be quite far from the targeted SCC-DFTB energy. In the course of the model development, we of course investigate the possibility to proceed as in SCC-DFTB with the ML charge inference only used as a starting point and then minimizing the energy defined by eq. 6, with the charge depending explicitly on the density matrix. In doing so, we use a first order algorithm as implemented in standard DFTB codes[32,33] and the output charge results from a steepest descent along the energy gradient with respect to orbital coefficients. Obviously, the first order algorithm gives no control of the length of such a displacement, and the output charge can be substantially different from the input charge, unless the inferred ML charges are very nearly identical to the converged SCC charges. We can overcome this difficulty by using the orbital rotation algorithm presented in the next section 2.3, at the expense of more time consuming second order algorithm. Moreover, if we content ourselves of one single diagonalization when starting from the SCC-DFTB, the sequence of energies produced along a deformation path does not coincide exactly with the minimum energy in a variational sense. At some points, the small deviation from SCC charge produces a Hamiltonian $h(q)$ such that the eigenfunctions and eigenenergies do not vary continuously along the path. Such a situation is quite likely when stretching a bond, when the antibonding and bonding orbitals merge to produce a localized orbital on the stretched atom. In other words, we observed diabatic orbital ordering changes along a deformation path, which introduces energy jumps when using the definition (6). We circumvented the problem by defining alternatively the energy with the help of equation 7 below:

$$E = \sum_{\mu\nu} \rho_{\mu\nu} h_{\mu\nu}(\mathbf{q}) + \frac{1}{2} \sum_{AB} q_A \gamma_{AB} Q_B + Q_A \gamma_{AB} q_B - \frac{1}{2} \sum_{AB} q_A \gamma_{AB} q_B \tag{7}$$

where $q_A - Q_A = \sum_{\mu\nu} \rho_{\mu\nu} S_{A,\mu\nu}$ by definition of the ionic charge Q_A. We obtain the machine learning energy by using the machine learning charges for each $\{q_A\}$ in this expression. Such an algorithm warrants the energy continuity, because the eigenorbitals now correspond exactly to the energy minimum.

Since the density matrix ρ results from the diagonalization of $h(\mathbf{q})$, this choice corresponds

to the minimum energy for the first term. In other word, we minimize the energy (7) for the value of the parameters q_A given by the machine learning algorithm. The first term is simply the weighted sum of the orbital energies ϵ_n resulting from the diagonalization of $h(\mathbf{q})$, $\sum_n \omega_n \epsilon_n$, where the weight ω_n is the occupation of the molecular orbital n. The expression (7) of the energy is equivalent to the SCC-DFTB energy when self-consistent charges are used. This expression is convenient, as it allows us to use the Helmann-Feynmann theorem to get the derivative of the first term without evaluating explicitly the density matrix derivative.

We can therefore obtain the forces as the energy derivatives with respect to the atomic coordinates, provided we know the charge derivatives with respect to the atomic coordinates. Since the ACS functions are themselves derivable quantities with respect to atomic coordinates, the ML charges are also derivable quantities. Using the definition 4, we obtain the energy derivative with respect to the atomic coordinate X:

$$\begin{aligned}\frac{\partial E}{\partial X} &= \sum_{\mu\nu}\left(\rho_{\mu\nu}\frac{\partial h_{\mu\nu}(\mathbf{0})}{\partial X} - \sigma_{\mu\nu}\frac{\partial S_{\mu\nu}}{\partial X}\right) + \frac{1}{2}\sum_{\mu\nu}\rho_{\mu\nu}\sum_{AB}\frac{\partial}{\partial X}\left(q_A\gamma_{AB}S_{B,\mu\nu} + S_{A,\mu\nu}\gamma_{AB}q_B\right) \\ &+ \frac{1}{2}\sum_{AB}\frac{\partial}{\partial X}\left(q_A\gamma_{AB}Q_B + Q_A\gamma_{AB}q_B\right) - \frac{1}{2}\sum_{AB}\frac{\partial}{\partial X}\left(q_A\gamma_{AB}q_B\right) \end{aligned} \tag{8}$$

where $\sigma_{\mu\nu} = \sum_n \omega_n \epsilon_n c_{n\mu} c_{n\nu}$ is the energy weighted density matrix, with $c_{n\mu}$ the expansion coefficients of the orbital n in the atomic basis μ. We can reorganize a bit the above expression. We first introduce the electronic charge resulting from the diagonalization:

$$p_A = \sum_{\mu\nu}\rho_{\mu\nu}S_{A,\mu\nu} \tag{9}$$

and we define the potential derivative with the help of the symmetry property $\gamma_{AB} = \gamma_{BA}$:

$$\begin{aligned}\frac{\partial V_{\mu\nu}}{\partial X} &= \frac{1}{2}\sum_{AB}\left(q_A\gamma_{AB}\frac{\partial S_{B,\mu\nu}}{\partial X} + q_B\gamma_{BA}\frac{\partial S_{A,\mu\nu}}{\partial X}\right) \\ &= \sum_{A,B} q_A\gamma_{AB}\frac{\partial S_{B,\mu\nu}}{\partial X} \end{aligned} \tag{10}$$

With these two definitions, we obtain eventually:

$$\begin{aligned}\frac{\partial E}{\partial X} &= \sum_{\mu\nu}\left(\rho_{\mu\nu}\left(\frac{\partial h_{\mu\nu}(\mathbf{0})}{\partial X} + \frac{\partial V_{\mu\nu}}{\partial X}\right) - \sigma_{\mu\nu}\frac{\partial S_{\mu\nu}}{\partial X}\right) + \frac{1}{2}\sum_{AB}\gamma_{AB}(p_A + Q_A - q_A)\frac{\partial q_B}{\partial X} \\ &+ \frac{1}{2}\sum_{AB}\frac{\partial\gamma_{AB}}{\partial X}\left(q_A(p_B + Q_B) + q_B(p_A + Q_A) - q_Aq_B\right) \end{aligned} \tag{11}$$

When the ML charges are identical to the SCC charges, we have $q_A = p_A + Q_A$. Thus the second summation cancels out and the last summation simplifies to $\sum_{AB}\frac{\partial\gamma_{AB}}{\partial X}q_Aq_B$, so that we recover the SCC-DFTB expression of the energy derivative, as expected.

To make the above expression practical, we need a closed form expression of the ML charge derivatives. The expression depends on the machine learning parameters and is detailed in section 3.

2.3 Charge guess as input of SCC calculation

It is definitely desirable to use the ML charge inference to speed up the standard SCC-DFTB calculation. In such a case, the whole machinery of existing SCC-DFTB codes would be available to obtain energy, forces and other properties, and there would be no need for an alternative energy definition like eq. 7. We evaluate such a possibility in this section.

Obviously, the ML charges do not match exactly the charges resulting from the complete SCC cycles, and their improvement is necessary to achieve self consistency. Unfortunately, the very first steps of the Broyden algorithm used in standard SCC-DFTB packages like

DFTB+[32] or deMon-Nano[14] do not take advantage of the improved charge guess. To circumvent this problem, we use a different algorithm based on molecular orbital rotation to minimize the energy. The starting point of the minimization is the set of molecular orbitals obtained by diagonalization of the SCC-DFTB Hamiltonian built with the ML charge guess $\mathbf{q}$. The gradient and Hessian with respect to orbital rotation R are then calculated to determine the orbital rotation parameters R_{mn}. The expression of energy derivatives with respect to rotation parameters is given in supporting information section 1. Since the minimization algorithm is based on a second order approximation of the energy surface with respect to R, a new approximate set of orbitals is generated by minimization of the approximate energy. This new set of orbitals is then used to start a new cycle of orbital rotation until the energy reaches a minimum. There are various ways to exploit the information contained in the second order derivative of the energy. We simply use here an augmented Hessian method. It is extremely efficient when the starting point is inside the quadratic area for the energy surface, and a few iterations are sufficient to achieve convergence up to machine accuracy. Moreover, such an algorithm is often more stable than first order algorithm based on iterative diagonalization of the DFTB Hamiltonian. The efficiency of the algorithm is clearly related to the cluster and the accuracy of the input charge. In some cases, 1 or 2 cycles are sufficent to reach convergence. This is systematically less with our augmented Hessian algorithm than with the DFTB+ package standard algorithm which generally requires a minimum of 10 cycles.

The drawback of the method is the size of the full Hessian in the space of orbital rotation parameters, which scales unfavorably with the system size. Despite a smaller number of cycles, the number of algebraic operations in each cycle makes a single step often more demanding than a single diagonalization. In the present work, we do not attempt to improve the second order algorithm. Nevertheless, it is clear that improved first order algorithm as suggested by Challacombe[34] or various approximate second order methods as discussed for example in literature[35–39], combined with the cheap SCC-DFTB Hessian diagonal as a

preconditioner, could reduce significantly the computation time.

We analyze how the deviation from the SCC solution affects the number of necessary cycles to reach convergence. To realize this, we add a small variation randomly sampled from a Gaussian distribution to the converged SCC-DFTB charges. A standard deviation of 10^{-1} unit of charge places the system charges relatively far out of the quadratic area where our second-order algorithm converges efficiently. On the contrary, as soon as the average deviation is less than 10^{-3}, very few cycles, typically 1 or 2, are necessary to reach convergence. With our present ML algorithm, the charge deviation is typically 10^{-2}, which is good enough for our purpose. However, it is clear that an improvement of the charge guess could improve substantially the computational efficiency.

3 Machine learning parameterization

The efficiency of the model presented in the above section depends critically on its ability to reproduce the atomic charge of the reference SCC-DFTB calculation by means of the Lagrangian defined in equation 1. The machine learning process consists in obtaining the values of the $\{\chi_A\}$, each χ_A being obtained from a corresponding set of ACSF $\{g_i\}_A$. The $\boldsymbol{\chi}$ are calculated through a dedicated High Dimensional Neural Network (HDNN) for each element, either C or Si in the present case, so that the difference between the reference and model charges is minimized for a large training set of reference geometries. The choice of ACSF is quite common for HDNN[26,40]. We took them from Himanen *et al.*[41] and we give the list of the parameters used in the present work in supporting information section 2. Alternative choices of rotation, translation and permutation invariant parameters are possible[42–44].

The training process was done with the PyTorch[45] package. The HDNN is made of 3 successive hidden-layers of rank 72, 72 and 34. The choice of the cutoff function between each layer is important to ensure the derivability of the charge with respect to the atomic positions. In the present work, we choose the modified SoftPlus[46] functions $f(x) = \log(1 + e^{-\beta x})$ with

a parameter $\beta = 100$. This kind of function performs better than the hyperbolic tangent often used in ML. They are twice differentiable and correct to some extent for the vanishing gradient problem, which may happen in the minimization process. To summarize, our HDNN consists in a chain of applications $\boldsymbol{\chi} = \phi_4 \circ \phi_3 \circ \phi_2 \circ \phi_1 \circ \phi_0(\mathbf{g})$ defined as follows:

$$\begin{aligned} \chi_i &= f(y_i^4) = \phi_4(\mathbf{g}^4) & y_i^4 &= \sum_k w_{ik}^4 g_k^4 + b_i^4 \\ g_i^{n+1} &= f(y_i^n) = \phi_n(\mathbf{g}^4) & y_i^n &= \sum_k w_{ik}^n g_k^n + b_i^n \\ g_i^0 &= g_i \end{aligned} \tag{12}$$

The optimization of the ML parameters w_{ik}^n and b_i^n is done by means of the Adam method[47] with a learning rate of 10^{-4}. For the ML training, we generate the atomic charges according to Mulliken definition for a series of clusters computed using the SCC-DFTB method[13] with the DFTB+ package[32] and the Matsci03 parameterization[25]. We do not attempt to optimize the DFTB parameters for charged systems and we expect the average atomic charge of 0.10 to be sufficiently small for the neutral parameterization to be valid.

The size and shape of the clusters were chosen randomly from a spherical cut in a SiC $P6_3mc6H$ lattice, with different positions for the cluster center. For each cluster so obtained, we first minimize the energy by means of a steepest descent algorithm for 100 iterations. Then we perform a molecular dynamics simulation in the micro-canonical ensemble with a time step of 2 fs over a few 100 iterations. The initial conditions for this dynamics are obtained by sampling a Maxwell-Boltzmann distribution of velocity for a temperature of 500 K. This choice of temperature was done to probe sufficiently large geometric deformation of the cluster. We also enhance the training set by sampling randomly the chemical nature of the surface atoms and then repeat the above algorithm. We keep a few configurations randomly chosen along each trajectory.

Finally, the training set is made of 80% of the selected geometries. The test and validation

sets are made of 10% each. The whole set of clusters is summarized in table 1.

Table 1: Number of SiC cluster structures and associated number of atoms grouped by range of size.

Type	Size range	Structure	Atoms
cluster	0-10	15307	87795
	11-80	52140	2114304
	81-300	29338	4710602
	301-800	242	89216
Total		97269	7001917

It is of course important that the ML charges are simply derivable quantities with respect to the atomic position, so that the forces could be obtained at a cheap numerical cost. To achieve our goal, we express the vector $\mathbf{p}$ as a combination of atomic charges $\mathbf{q}$ with the Lagrange parameter λ:

$$\mathbf{p} = A^{-1}\mathbf{\Theta} \quad \text{with} \quad \mathbf{p} = \begin{pmatrix} \mathbf{q} \\ \lambda \end{pmatrix} , \mathbf{\Theta} = \begin{pmatrix} -\boldsymbol{\chi} \\ Q_{tot} \end{pmatrix} \text{ and } A = \begin{pmatrix} \boldsymbol{\gamma} & 1 \\ 1 & 0 \end{pmatrix} \tag{13}$$

where Q_{tot} is the total charge of the cluster and $\boldsymbol{\gamma}$ matrix is the same as defined in SCC-DFTB formalism[13,14]. The derivative of $\mathbf{p}$ with respect to an atomic coordinate X reads:

$$\begin{aligned} \frac{\partial \mathbf{p}}{\partial X} &= \frac{\partial A^{-1}}{\partial X}\mathbf{\Theta} + A^{-1}\frac{\partial \mathbf{\Theta}}{\partial X} \\ &= A^{-1}\frac{\partial A}{\partial X}A^{-1}\mathbf{\Theta} + A^{-1}\frac{\partial \mathbf{\Theta}}{\partial X} \end{aligned} \tag{14}$$

The numerical derivative of the matrix A requires the analytic derivative of $\boldsymbol{\gamma}$ as given by Elstner *et al.*[13]. There is one matrix derivative for each coordinate. However, only one single row and one single column of A depend on a given coordinate X, and this remains a cheap computation.

In the second term, $\frac{\partial \mathbf{\Theta}}{\partial X}$ is composed of the derivative of the neural network with respect to the ACSF vector $\mathbf{g}$ and their derivatives. It is obtained by chain rule differentiation of

equation (12) to obtain another chain of applications. The derivative chain is merely the original chain with f changed for f'. It reads explicitly:

$$\begin{aligned} \frac{\partial \chi_i}{\partial X} &= f'(y_i^4) \sum_k w_{ik}^4 \frac{\partial g_k^4}{\partial X} \\ \frac{\partial g_i^{n+1}}{\partial X} &= f'(y_i^n) \sum_k w_{ik}^n \frac{\partial g_k^n}{\partial X} \\ \frac{\partial g_i^0}{\partial X} &= \frac{\partial g_i}{\partial X} \end{aligned} \tag{15}$$

where f' is the derivative of the function f and the variables y_i^n are defined in equation (12).

We initiate the derivative chain with the derivatives of the ACSF. There is one such chain for each atom coordinate. There are thus as many HDNN evaluations as there are atoms times the number of coordinates, *i.e.* $3N^2$ chains for N atoms. The derivative f' is quite simple and the algebraic operations are fast for the small matrices of the network, so that the HDNN derivation itself is not the dominant part of the computation time, which is dominated by the evaluation of the ACSF derivatives. The whole algorithm for ML-DFTB forces is competitive with respect to the SCC-DFTB forces, which does not require the atomic charge derivatives.

4 Relaxed geometry of charged SiC clusters

As a first endeavour, we have relaxed a few cluster geometries with ML-DFTB, SCC-DFTB and also DFT as a reference. The DFT computation was done by using PBE functional and default HGH pseudo-potential with the BigDFT package[48]. We consider three different kinds of geometry corresponding to bulk, cage and segregated atomic arrangements.

We have first relaxed a $(SiC)_{37}$ cluster with a total charge $Q_{tot} = +8$. The initial positions were obtained from a cut of the cluster in a 6H SiC crystal. Since the forces are available in ML-DFTB, the optimization by means of a BFGS method is quite fast. The relaxed

geometries are illustrated in figure 3. The difference between the relaxed geometries can barely be observed in the figure. We have therefore characterized the similarity between two structures as the normalized dot product of the ACSF of the two structures, $\mathbf{g} \cdot \mathbf{g}'/gg'$. These quantities are depicted by the colored matrix on the left hand side. The dark blue with a value of 1 corresponds to perfect identity, *i.e.* $\mathbf{g} = \mathbf{g}'$, while lighter colors correspond to a larger difference. We observe an excellent agreement between the ML-DFTB and SCC-DFTB. The agreement of both SCC-DFTB and ML-DFTB with DFT is also fair. The clustering of the C atoms as C_2 pairs is well reproduced by both parametric methods. On the other hand, the main difference comes from the outermost Si atoms (top left of the cluster).

Cage-like neutral clusters of SiC have been investigated by Patrick *et al.*[49]. These kinds of structures are quite stable, in particular for the $(SiC)_{12}$ cluster, though they do not correspond to the lowest energy isomer[50,51]. We present here the relaxed geometry for $(SiC)_{12}$ with a total charge $Q_{tot} = +2$. The geometries are shown in figure 4. We observe an excellent agreement between ML-DFTB and SCC-DFTB. The agreement with DFT is fair also, though the DFT structure is more compact. The origin of the difference lies obviously the DFTB parameters, which have been optimized for bulk rather than for cluster structure.

Finally, we present in figure 5 a segregated cluster of $(SiC)_{12}$ for total charge $Q_{tot} = +2$. According to DFT calculation, such segregated structures, with the carbon atoms on one side and the silicon atoms on the other side, are typical of small neutral SiC clusters[50]. The isomer energy ordering is however sensitive to the functional choice. More elaborated wave function calculations predict a more symmetric closo structure to be the lowest energy isomer[51]. The B3LYP and PBE functionals reproduce this isomer ordering. The starting geometry for our relaxation corresponds to the isomers obtained by DFT calculation for neutral $(SiC)_{12}$ cluster[50]. In the case of charged clusters with $Q_{tot} = +2$ our own DFT calculation with PBE functional gives the closo isomer to be the lowest energy isomer, while the segregated isomer depicted in figure 5 is 1.2 eV above. The SCC-DFTB reproduces the correct ordering for

these two isomers. However, the ML-DFTB fails to reproduce this ordering. This is a clear indication that the training set lacks of non-stoichiometric structures, *i.e.* carbon-rich or silicon-rich structures and the atomic charges predicted for such structures deviate from the reference SCC-DFTB charges. Nevertheless, the agreement between ML-DFTB and SCC-DFTB regarding the geometry is excellent. The agreement with DFT is fair also and the global shape is faithfully reproduced by the parameterized models. Most of the difference comes from the fine positioning of the Si atoms, a more accurate parameterization of the DFTB parameters might improve the agreement.

We have finally analyzed the deviation from the average atomic charge used in the training set. We have computed the mean error in the predicted charge and in the resulting energy when computing these quantities with the ML-DFTB model for an average charge per atom varying by ±20 percent. The calculation was done for 250 randomly chosen clusters with a size distributed between 10 and 200 atoms. The results are plotted in figure 6. Regarding the atomic charge distribution, the mean value deviates more or less quadratically from the reference value of 0.01 at $\bar{q} = 0.10$ to reach 0.03 at $\bar{q} = 0.08$ and $\bar{q} = 0.12$. The variation does not depend significantly on the cluster size. Regarding the energy, the relative variation is much smaller than for the charge, except for the point at $\bar{q} = 0.115$ where it becomes as large as 12 meV, *i.e.* 50% more than the standard deviation of 8 meV observed for the other charges. Thus, we believe our approach to be valid in a range of an average charge of ±10 percent around the value used to set the ML parameters. Increasing the range of charge validity would certainly be possible if a broader training ensemble, made of different reference charges, was used.

5 Stability of charged SiC clusters

For charged clusters and also for sharp tips, the stability is governed by ion field emission. A key quantity in this matter is the energy barrier an atom has to overcome to be released

out of the cluster. As an illustration of the method, we give here an example of such barriers for C and Si emission from a $(SiC)_{37}$ cluster with a total charge $Q_{tot} = +7$. The geometry of the cluster was optimized from a cut in the bulk structure, but we do not attempt to search for the lowest energy isomer. Our purpose here is only to check the consistency of the ML-DFTB with respect to SCC-DFTB regarding the energy barrier. The C and Si atoms were chosen to be the less bound surface atoms for each element. As a test case, we simply move the atom out of the cluster along the path defined by its initial position and the center of mass of the cluster.

The figure 7 presents the energy obtained with our model and a full SCC-DFTB calculation along such a path for C atom. We have also plotted, on the lower panel, the charge of the stretched atom with respect to the displacement length. The abscissa is the distance between the atom and the centre of mass of the cluster. Regarding the charge, the agreement of the ML-DFTB model with the reference SCC-DFTB calculation is rather good, in particular in the long range limit, dominated by the Coulomb interaction between the stretched carbon atom and the remaining cluster.

We observe larger deviation at short distances around 6 Å, corresponding to the equilibrium position of the stretched atom on the cluster surface. In this range, the model energy is deeper by about 1 eV, which is an average error of about 13 meV/atom. This is typically the precision reached for energy prediction by machine learning models. For instance in their work R. Haffizi *et al.*[52] report that this error amount to 22 meV/atom, while T.W. Ko *et al.*[3] reach 7.3 meV/atom for their best model with respect to this criterion.

Most significantly, the shape of the dissociation energy barrier is quite well reproduced all along the dissociation path. For instance in figure 7, the measured energy barriers with both method amount to 5.08 eV; the relative error of the ML-DFTB approximation amounting to $1.5 \cdot 10^{-3}$.

In the figure 8, we have plotted the result of stretching for a Si atom. The agreement is as good as in this case of C stretching. We also observed a small systematic energy shift of

the order of 1 eV, but the shape of the barrier is also faithfully reproduced and the atomic charge matches quite well the reference charge.

In the present case, the height of the barrier is of the order of 5.5 eV for both elements. This is obviously an upper dissociation limit because the cluster atoms are frozen at their initial equilibrium position. We observe that the carbon atom carries out more charge than the silicon atom. Accordingly, the C atom stretching lowers the energy more than the Si atom stretching. From a thermodynamic point of view, C atom emission is therefore preferable.

These results are gratifying as they offer the possibility to study dissociation barriers with confidence for much larger clusters and with more accurate methods such as nudge elastic band, molecular dynamics simulation or Monte Carlo methods.

6 Conclusion

We have used a machine learning method to infer the atomic charges used in SCC-DFTB calculation. The quality of the charge inference is of the order of $\pm 10^{-2}$ unit of charge. We have shown that this is sufficiently good to obtain meaningful energies at the cost of one single tight-binding calculation. This is a significant saving with respect to standard SCC-DFTB calculation, for which a large number of SCC iterations are necessary to achieve convergence. As shown in supporting information section 3 the saving is of the order of the number of diagonalizations necessary to reach convergence in SCC-DFTB calculation, *i.e.* typically 30 to 100 cycles (or even up to a few hundred for a large charged cluster with several equivalent atomic positions). In our ML-DFTB approach, the computational effort is transferred to the training part of the machine learning. Moreover, there are some cases for which the SCC-DFTB convergence is poor or even fails, most often because of near degeneracy of the orbital energy around the Fermi level. The ML-DFTB does not suffer from this drawback, and a safe alternative algorithm based on orbital rotation can be used to exploit efficiently the ML charge as a first guess.

Unlike previous development of HDNN oriented toward prediction of energy, the ML-DFTB model gives access to energy, forces and atomic charges, and hence to the electronic density and the related properties. The force evaluation is analytical, which guarantees a fast evaluation with respect to finite difference scheme. The ML-DFTB model presented here opens new possibilities to study large molecules, atomic clusters or solids with an accuracy close to that of SCC-DFTB. The method is obviously not restricted to SiC clusters and numerous sets of DFTB parameters for other elements are available to build up a database for charge inference. Thus, the large possibilities of SCC-DFTB to investigate many different materials of various chemical nature may be transferred to ML-DFTB. Regarding the accuracy, robustness and transferability of the model, it is clear that the ML-DFTB inherit from the limitations of the SCC-DFTB. If more accuracy is necessary, in particular for dissociation barrier, it is certainly possible to improve the DFTB parameterization in several ways, as discussed in the literature[16–18].

The present work was oriented toward charged cluster studies, with field emission in mind. The training set was thus build with charged clusters having an average charge per atom of 0.10 unit of charge. Nevertheless, we have seen that the ML-DFTB reproduces quite faithfully the SCC-DFTB results for clusters with smaller and larger average charge around the chosen training charge of 0.10. It is obviously possible to enhance the training set to extent the charge range, including the important case of neutral clusters. For further work on charged clusters, it may be interesting to consider a model based on DFTB3[53] and more elaborated charge definition, for instance one of those cited in the overview of charge models given by Marenish *et al.*[54], instead of the standard Mulliken definition. A ML-DFTB3 method would prove especially useful as the SCC charge convergence in DFTB3 is much more tricky than the original one[13].

The possibility to investigate systems made of thousands of atoms opens new possibilities of simulation in the field of APT. In particular, the emission barriers and the preferred emission sites could be investigated by means of nudge elastic band. It is also possible to perform

dynamics to investigate atom migration or surface reconstruction following atom emission. While in principle feasible with more accurate DFT methods, the current development of DFT codes does not allow us to perform such a research program routinely. In the same topic of ion field emission, the ML-DFTB method is also attractive for the investigation of charged cluster stability.

Acknowledgement

We thanks the french Labex EMC3 for supporting the BreakinGAP project. We also thank the Normandy region for supporting the AMODACCF project.

7 Associated Content

As supporting information, the reader will find the three following items. First the literal computation of first and second order derivatives of the DFTB2 hamiltonian with respect to rotations within the orbitals space. Second the ASCF parameters used to describe atomic environements for the machine learning charge prediction model we used. And third the timings improvement we have measured for SiC clusters with our prototypical implementation of ML-DFTB.

References

(1) Snyder, J. C.; Rupp, M.; Hansen, K.; Müller, K.-R.; Burke, K. Finding Density Functionals with Machine Learning. *Phys. Rev. Lett.* **2012**, *108*, 253002.

(2) Behler, J.; Parrinello, M. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Phys. Rev. Lett.* **2007**, *98*, 146401.

(3) Ko, T. W.; Finkler, J. A.; Goedecker, S.; Behler, J. General-Purpose Machine Learning Potentials Capturing Nonlocal Charge Transfer. *Accounts of Chemical Research* **2021**, *54*, 808–817, PMID: 33513012.

(4) Schütt, K. T.; Arbabzadah, F.; Chmiela, S.; Müller, K. R.; Tkatchenko, A. Quantum-Chemical Insights from Deep Tensor Neural Networks. *Nature Communications* **2017**, *8*, 13890.

(5) Schütt, K. T.; Sauceda, H. E.; Kindermans, P.-J.; Tkatchenko, A.; Müller, K.-R. SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics* **2018**, *148*, 241722.

(6) Unke, O. T.; Meuwly, M. PhysNet: A Neural Network for Predicting Energies, Forces, Dipole Moments and Partial Charges. *Journal of Chemical Theory and Computation* **2019**, *15*, 3678–3693.

(7) Zubatyuk, R.; Smith, J. S.; Nebgen, B. T. Teaching a neural network to attach and detach electrons from molecules. *Nature Communications* **2021**, *12*, 4870.

(8) Deringer, V. L.; Bartók, A. P.; Bernstein, N.; Wilkins, D. M.; Ceriotti, M.; Csányi, G. Gaussian Process Regression for Materials and Molecules. *Chemical Reviews* **2021**, *121*, 10073–10141, PMID: 34398616.

(9) Ghasemi, S. A.; Hofstetter, A.; Saha, S.; Goedecker, S. Interatomic potentials for ionic

systems with density functional accuracy based on charge densities obtained by a neural network. *Physical Review B* **2015**, *92*, 045131.

(10) Behler, J.; Parrinello, M. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Physical Review Letters* **2007**, *98*, 146401.

(11) Ko, T. W.; Finkler, J. A.; Goedecker, S.; Behler, J. A fourth-generation high-dimensional neural network potential with accurate electrostatics including non-local charge transfer. *12*, 398.

(12) Faber, F. A.; Hutchison, L.; Huang, B.; Gilmer, J.; Schoenholz, S. S.; Dahl, G. E.; Vinyals, O.; Kearnes, S.; Riley, P. F.; von Lilienfeld, O. A. Prediction Errors of Molecular Machine Learning Models Lower than Hybrid DFT Error. *Journal of Chemical Theory and Computation* **2017**, *13*, 5255–5264, PMID: 28926232.

(13) Elstner, M.; Porezag, D.; Jungnickel, G.; Elsner, J.; Haugk, M.; Frauenheim, T.; Suhai, S.; Seifert, G. Self-consistent-charge density-functional tight-binding method for simulations of complex materials properties. *Physical Review B* **1998**, *58*, 7260–7268.

(14) Spiegelman, F.; Tarrat, N.; Cuny, J.; Dontot, L.; Posenitskiy, E.; Marti, C.; Simon, A.; Rapacioli, M. Density-functional tight-binding: basic concepts and applications to molecules and clusters. *Advances in Physics: X* **2019**, *5*, 1710252.

(15) Bannwarth, C.; Ehlert, S.; Grimme, S. GFN2-xTB—An Accurate and Broadly Parametrized Self-Consistent Tight-Binding Quantum Chemical Method with Multipole Electrostatics and Density-Dependent Dispersion Contributions. *Journal of Chemical Theory and Computation* **2019**, *15*, 1652–1671, PMID: 30741547.

(16) Li, H.; Collins, C.; Tanha, M.; Gordon, G. J.; Yaron, D. J. A Density Functional Tight Binding Layer for Deep Learning of Chemical Hamiltonians. *Journal of Chemical Theory and Computation* **2018**, *14*, 5764–5776.

(17) Huran, A. W.; Steigemann, C.; Frauenheim, T.; Aradi, B.; Marques, M. A. L. Efficient Automatized Density-Functional Tight-Binding Parametrizations: Application to Group IV Elements. *Journal of Chemical Theory and Computation* **2018**, *14*, 2947–2954.

(18) Bissuel, D.; Albaret, T.; Niehaus, T. A. Critical assessment of machine-learned repulsive potentials for the density functional based tight-binding method: A case study for pure silicon. *The Journal of Chemical Physics* **2022**, *156*, 064101.

(19) Echt, O.; Scheier, P.; Märk, T. D. Multiply charged clusters. *Comptes Rendus Physique* **2002**, *3*, 353–364.

(20) Johnston, R. *Atomic and Molecular Clusters*; Taylor & Francis, 2002.

(21) Märk, T. D. Cluster ions: Production, detection and stability. *International Journal of Mass Spectrometry and Ion Processes* **1987**, *79*, 1–59.

(22) Ferrando, R.; Jellinek, J.; Johnston, R. L. Nanoalloys: From Theory to Applications of Alloy Clusters and Nanoparticles. *Chemical Reviews* **2008**, *108*, 845–910, PMID: 18335972.

(23) Haberland, H. *Clusters of Atoms and Molecules Theory, Experiment, and Clusters of Atoms*; Springer-Verlag, 1994.

(24) Lefebvre, W.; Vurpillot, F.; Sauvage, X. *Atom probe tomography: put theory into practice*; Academic Press, 2016.

(25) Semi-relativistic, self-consistent charge Slater-Koster tables for density-functional based tight-binding (DFTB) for materials science simulations. `https://dftb.org/parameters/download/matsci/matsci-0-3-cc`.

(26) Behler, J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *The Journal of Chemical Physics* **2011**, *134*, 074106.

(27) Ndiaye, S.; Bacchi, C.; Klaes, B.; Canino, M.; Vurpillot, F.; Rigutti, L. Surface Dynamics of Field Evaporation in Silicon Carbide. *The Journal of Physical Chemistry C* **2023**, *127*, 5467–5478.

(28) Faraji, S.; Ghasemi, S. A.; Rostami, S.; Rasoulkhani, R.; Schaefer, B.; Goedecker, S.; Amsler, M. High accuracy and transferability of a neural network potential through charge equilibration for calcium fluoride. *Phys. Rev. B* **2017**, *95*, 104105.

(29) Rasoulkhani, R.; Tahmasbi, H.; Ghasemi, S. A.; Faraji, S.; Rostami, S.; Amsler, M. Energy landscape of ZnO clusters and low-density polymorphs. *Phys. Rev. B* **2017**, *96*, 064108.

(30) Rappe, A. K.; Goddard, W. A. Charge equilibration for molecular dynamics simulations. *The Journal of Physical Chemistry* **1991**, *95*, 3358–3363.

(31) Ongari, D.; Boyd, P. G.; Kadoglu, O.; Mace, A. K.; Keskin, S.; Smit, B. Evaluating Charge Equilibration Methods To Generate Electrostatic Fields in Nanoporous Materials. *Journal of Chemical Theory and Computation* **2019**, *15*, 382-401.

(32) Hourahine, B.; Aradi, B.; Blum, V.; Bonafé, F.; Buccheri, A.; Camacho, C.; Cevallos, C.; Deshaye, M. Y.; Dumitrică, T.; Dominguez, A.; Ehlert, S.; Elstner, M.; van der Heide, T.; Hermann, J.; Irle, S.; Kranz, J. J.; Köhler, C.; Kowalczyk, T.; Kubař, T.; Lee, I. S.; Lutsker, V.; Maurer, R. J.; Min, S. K.; Mitchell, I.; Negre, C.; Niehaus, T. A.; Niklasson, A. M. N.; Page, A. J.; Pecchia, A.; Penazzi, G.; Persson, M. P.; Řezáč, J.; Sánchez, C. G.; Sternberg, M.; Stöhr, M.; Stuckenberg, F.; Tkatchenko, A.; Yu, V. W.-z.; Frauenheim, T. DFTB+, a software package for efficient approximate density functional theory based atomistic simulations. *The Journal of Chemical Physics* **2020**, *152*, 124101.

(33) Scemama, A.; Renon, N.; Rapacioli, M. A Sparse Self-Consistent Field Algorithm and

Its Parallel Implementation: Application to Density-Functional-Based Tight Binding. *Journal of Chemical Theory and Computation* **2014**, *10*, 2344–2354, PMID: 26580754.

(34) Challacombe, M. A simplified density matrix minimization for linear scaling self-consistent field theory. *The Journal of Chemical Physics* **1999**, *110*, 2332–2342.

(35) Francisco, J. B.; Martínez, J. M.; Martínez, L. Density-based Globally Convergent Trust-region Methods for Self-consistent Field Electronic Structure Calculations. *Journal of Mathematical Chemistry* **2006**, *40*, 349–377.

(36) Thøgersen, L.; Olsen, J.; Yeager, D.; Jørgensen, P.; Sałek, P.; Helgaker, T. The trust-region self-consistent field method: Towards a black-box optimization in Hartree–Fock and Kohn–Sham theories. *The Journal of Chemical Physics* **2004**, *121*, 16–27.

(37) Schlegel, H. B.; McDouall, J. J. W. In *Computational Advances in Organic Chemistry: Molecular Structure and Reactivity*; Ögretir, C., Csizmadia, I. G., Eds.; Springer Netherlands: Dordrecht, 1991; pp 167–185.

(38) Bacskay, G. B. A quadritically convergent hartree-fock (QC-SCF) method. Application to open shell orbital optimization and coupled perturbed hartree-fock calculations. *Chemical Physics* **1982**, *65*, 383–396.

(39) Douady, J.; Ellinger, Y.; Subra, R.; Levy, B. Exponential transformation of molecular orbitals: A quadratically convergent SCF procedure. I. General formulation and application to closed-shell ground states. *The Journal of Chemical Physics* **1980**, *72*, 1452–1462.

(40) Pozdnyakov, S. N.; Willatt, M. J.; Bartók, A. P.; Ortner, C.; Csányi, G.; Ceriotti, M. Incompleteness of Atomic Structure Representations. *Phys. Rev. Lett.* **2020**, *125*, 166001.

(41) Himanen, L.; Jäger, M. O. J.; Morooka, E. V.; Federici Canova, F.; Ranawat, Y. S.;

Gao, D. Z.; Rinke, P.; Foster, A. S. DScribe: Library of descriptors for machine learning in materials science. *Computer Physics Communications* **2020**, *247*, 106949.

(42) Barnard, T.; Tseng, S.; Darby, J. P.; Bartók, A. P.; Broo, A.; Sosso, G. C. Leveraging genetic algorithms to maximise the predictive capabilities of the SOAP descriptor. *Mol. Syst. Des. Eng.* **2023**, *8*, 300–315.

(43) De, S.; Bartók, A. P.; Csányi, G.; Ceriotti, M. Comparing molecules and solids across structural and alchemical space. *Physical Chemistry Chemical Physics* **2016**, *18*, 13754–13769.

(44) Jäger, M. O.; Morooka, E. V.; Canova, F. F. Machine learning hydrogen adsorption on nanoclusters through structural descriptors. *npj Comput Mater* **2018**, *4*, 37.

(45) Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. 2017; 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA.

(46) Glorot, X.; Bordes, A.; Bengio, Y. Deep Sparse Rectifier Neural Networks. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. Fort Lauderdale, FL, USA, 2011; pp 315–323.

(47) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. 2017.

(48) Ratcliff, L. E.; Dawson, W.; Fisicaro, G.; Caliste, D.; Mohr, S.; Degomme, A.; Videau, B.; Cristiglio, V.; Stella, M.; D'Alessandro, M.; Goedecker, S.; Nakajima, T.; Deutsch, T.; Genovese, L. Flexibilities of wavelets as a computational basis set for large-scale electronic structure calculations. *The Journal of Chemical Physics* **2020**, *152*, 194110.

(49) Patrick, A. D.; Dong, X.; Allison, T. C.; Blaisten-Barojas, E. Silicon carbide nanostructures: A tight binding approach. *The Journal of Chemical Physics* **2009**, *130*, 244704.

(50) Song, B.; Yong, Y.; Hou, J.; He, P. Density-functional study of Si_nC_n (n = 10–15) clusters. *The European Physical Journal D* **2010**, *59*, 399–406.

(51) Byrd, J. N.; Lutz, J. J.; Jin, Y.; Ranasinghe, D. S.; Montgomery, J., John A.; Perera, A.; Duan, X. F.; Burggraf, L. W.; Sanders, B. A.; Bartlett, R. J. Predictive coupled-cluster isomer orderings for some Si_nC_m (m, n $\leq$ 12) clusters: A pragmatic comparison between DFT and complete basis limit coupled-cluster benchmarks. *The Journal of Chemical Physics* **2016**, *145*, 024312.

(52) Hafizi, R.; Ghasemi, S. A.; Hashemifar, S. J.; Akbarzadeh, H. A neural-network potential through charge equilibration for WS2: From clusters to sheets. *The Journal of Chemical Physics* **2017**, *147*, 234306.

(53) Gaus, M.; Cui, Q.; Elstner, M. DFTB3: Extension of the Self-Consistent-Charge Density-Functional Tight-Binding Method (SCC-DFTB). *Journal of Chemical Theory and Computation* **2011**, *7*, 931–948.

(54) Marenich, A. V.; Jerome, S. V.; Cramer, C. J.; Truhlar, D. G. Charge Model 5: An Extension of Hirshfeld Population Analysis for the Accurate Description of Molecular Interactions in Gaseous and Condensed Phases. *Journal of Chemical Theory and Computation* **2012**, *8*, 527–541, PMID: 26596602.

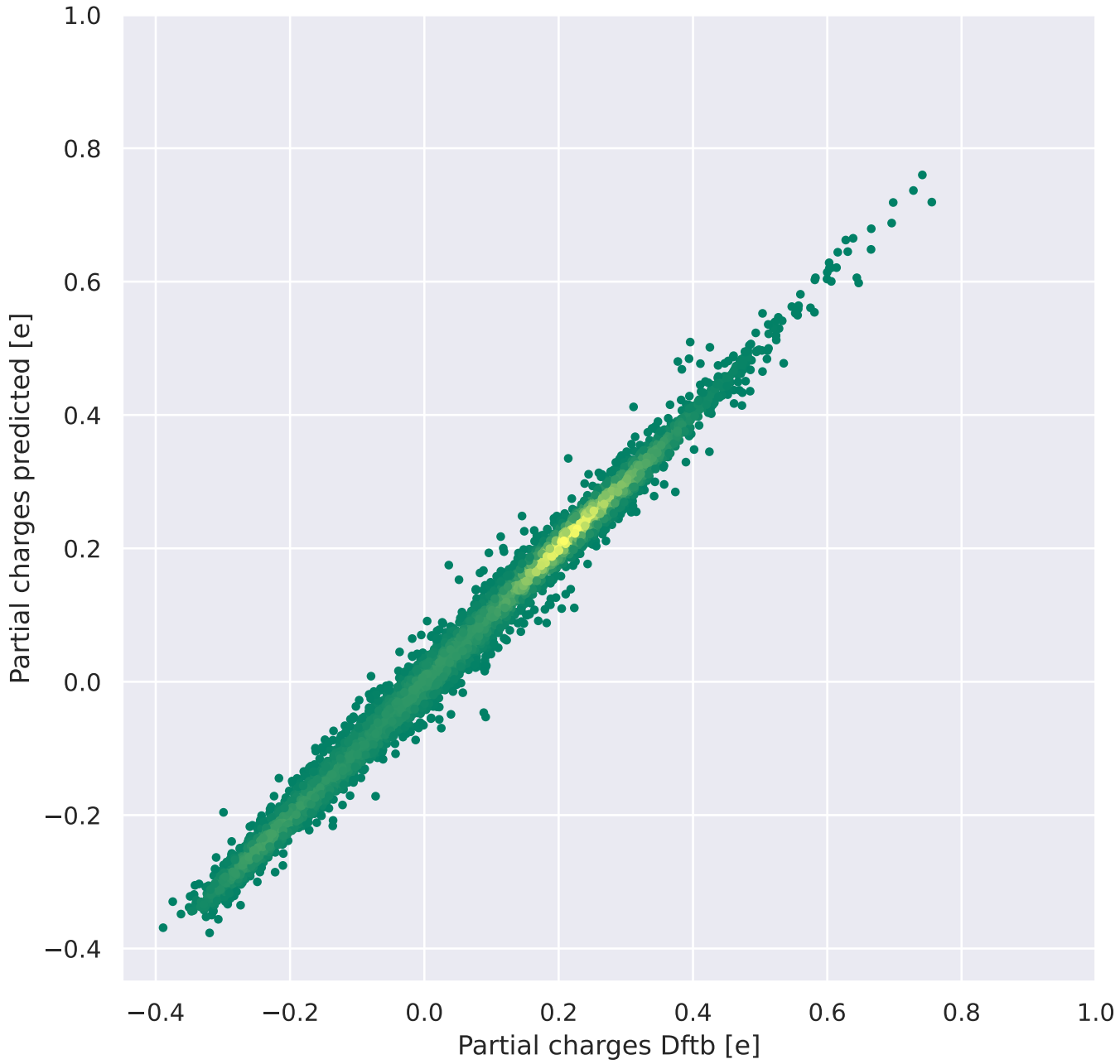

Figure 1: Correlation plot of the ML charges versus SCC charges for our training set with an average charge per atom $\bar{q} = 0.10$. The colour scale is indicative of the number of points at a given charge: the more yellow the colour, the higher the number of points.

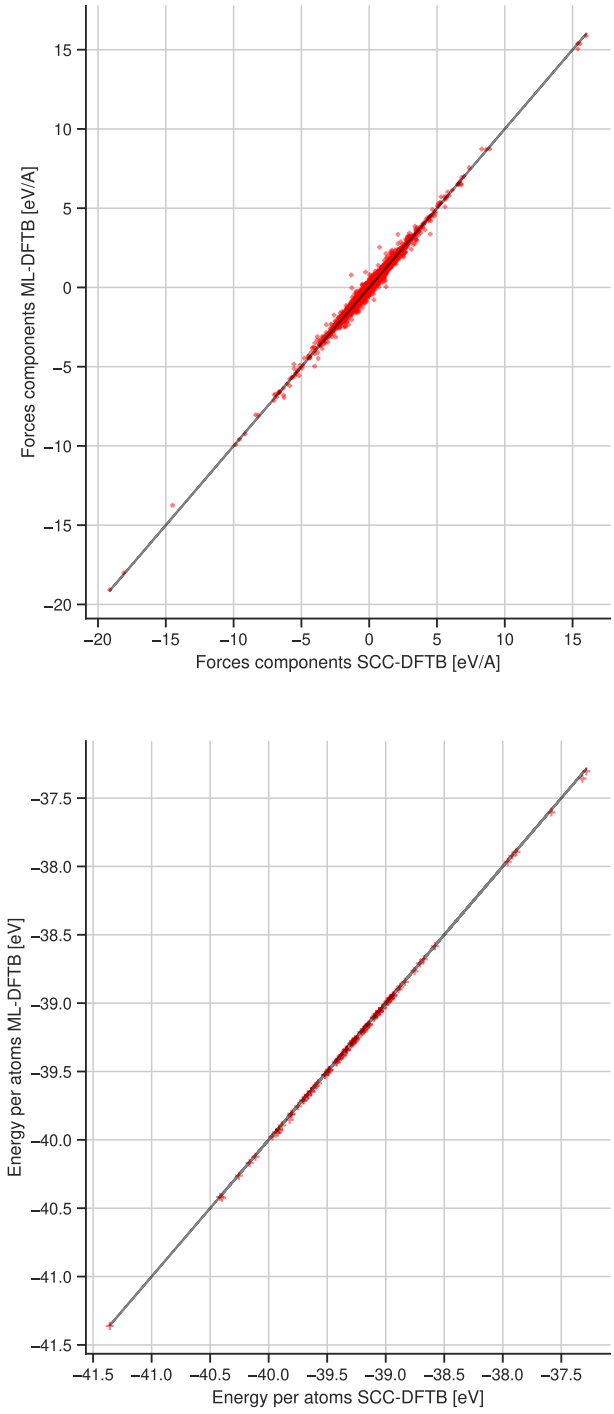

Figure 2: Correlation diagram of ML-DFTB versus SCC-DFTB, for energy and force components.

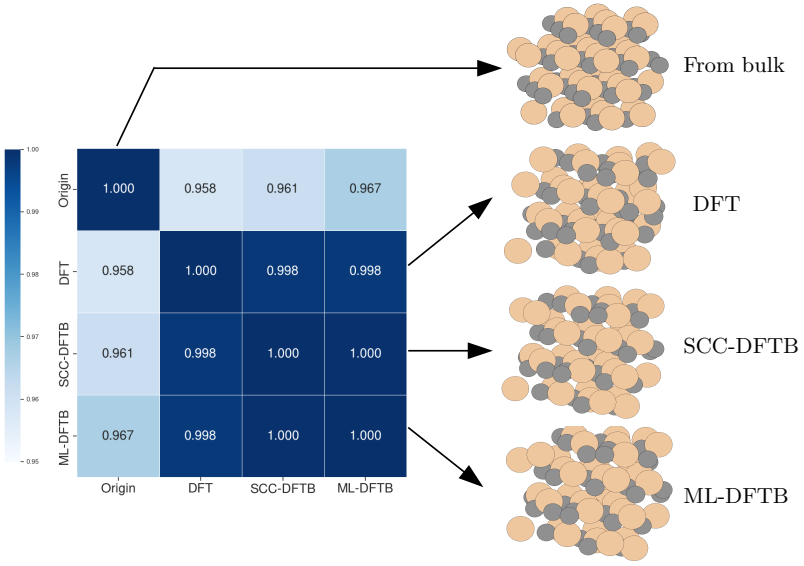

Figure 3: Comparison of the $(SiC)_{37}$ structures obtained from direct 6H crystal without relaxation, and the relaxed DFT, SCC-DFTB and ML-DFTB, with a total charge $Q = +8$. The colored matrix indicates the agreement between two structures. The perfect agreement corresponds to dark blue and the lighter the blue, the stronger the disagreement.

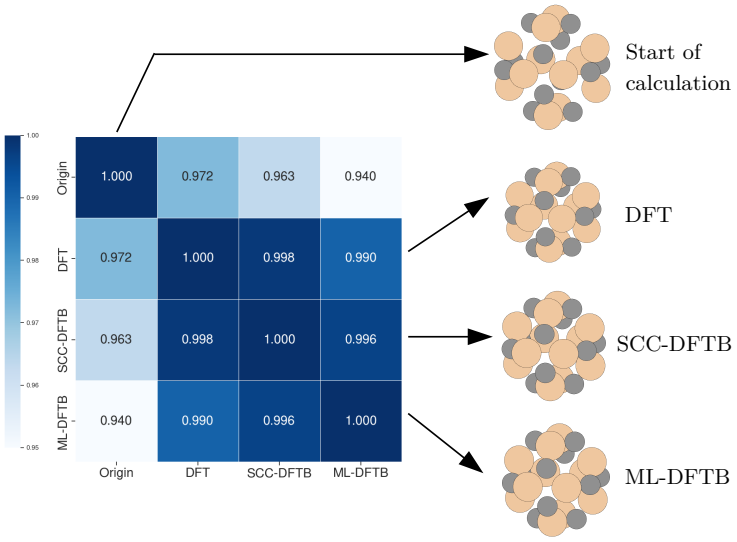

Figure 4: Comparison of the $(SiC)_{12}$ cage structures obtained from our initial guess, and the relaxed DFT, SCC-DFTB and ML-DFTB. The colored matrix indicates the agreement between two structures. The perfect agreement corresponds to dark blue and the lighter the blue, the stronger the disagreement.

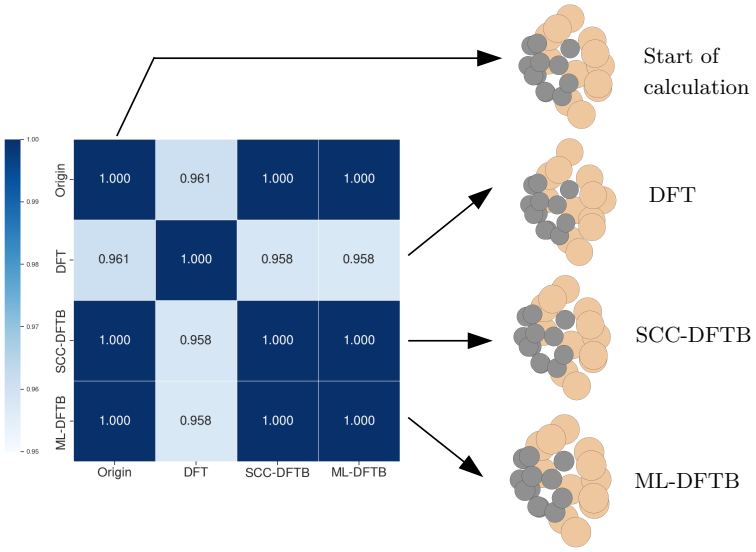

Figure 5: Comparison of the $(SiC)_{12}$ segregated structures obtained from our initial guess, and the relaxed DFT, SCC-DFTB and ML-DFTB. The colored matrix indicates the agreement between two structures. The perfect agreement corresponds to dark blue and the lighter the blue, the stronger the disagreement.

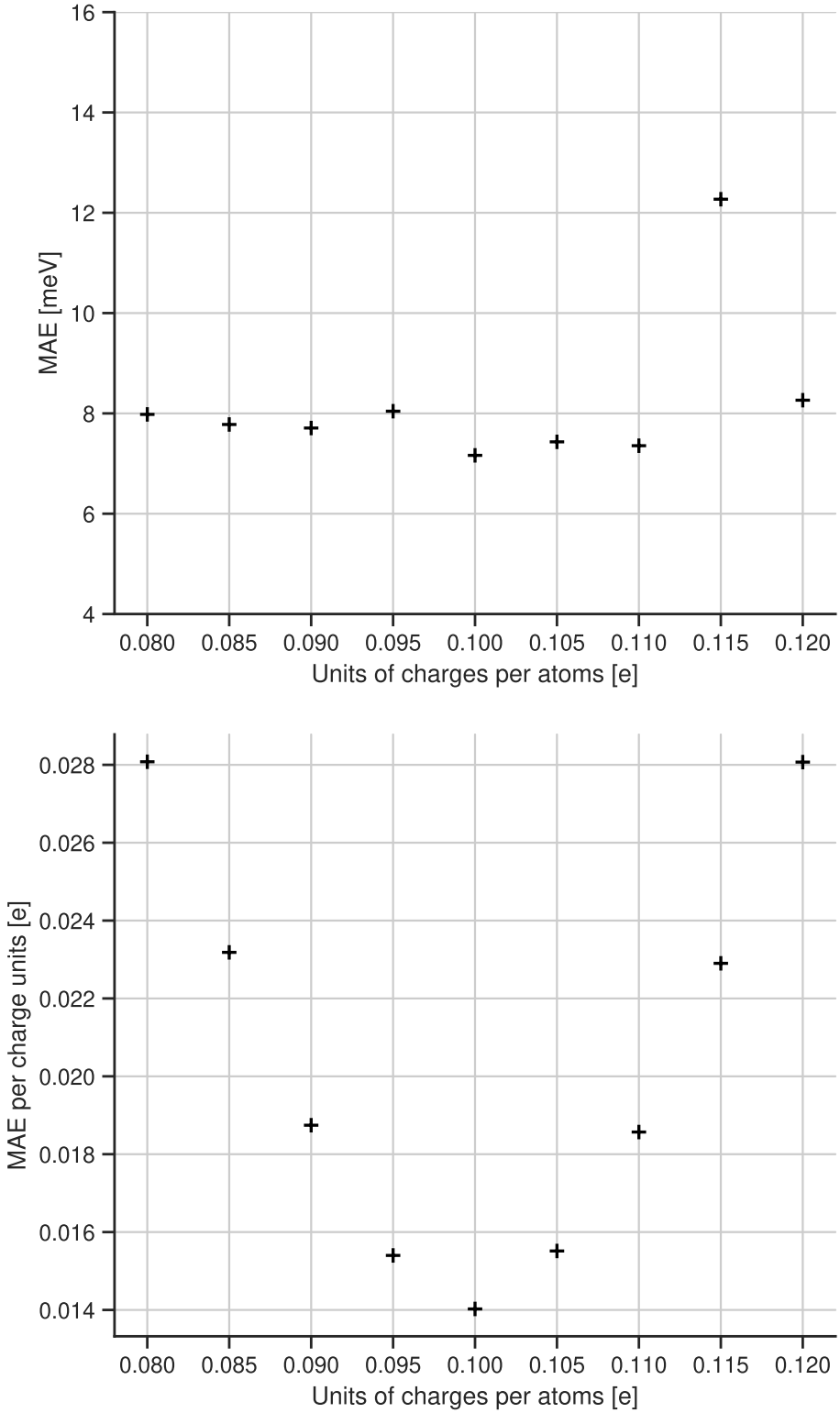

Figure 6: Mean absolute energy error (upper panel) and mean absolute charge error (lower panel) as a function of the charge constraint expressed as average atomic charge $\bar{q}$, for our training set characterized by $\bar{q} = 0.10$.

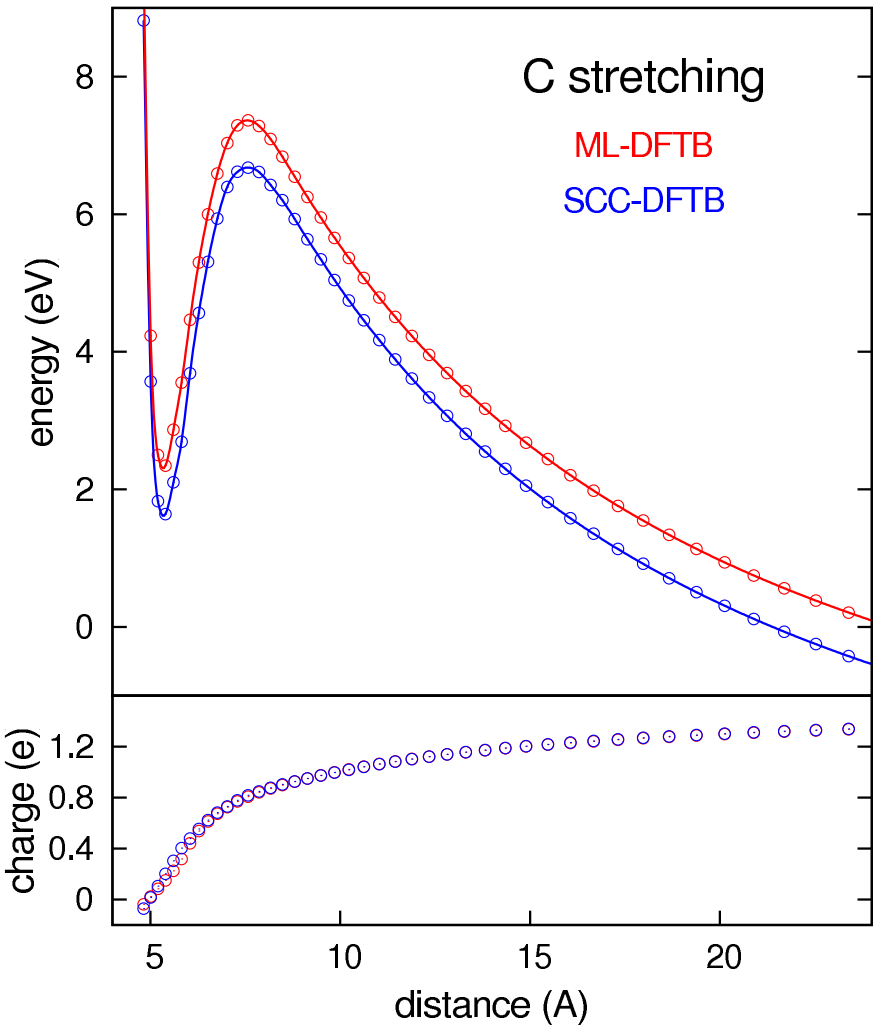

Figure 7: Energy and charge (respectively upper and lower panel) along a dissociation path for C atom stretching. The abscissa is the distance between the stretched atom and the centre of mass of the cluster. The small difference of about 1 eV between the ML-DFTB and reference potential energy is mainly a cumulative effect associated to tiny atomic charge differences.

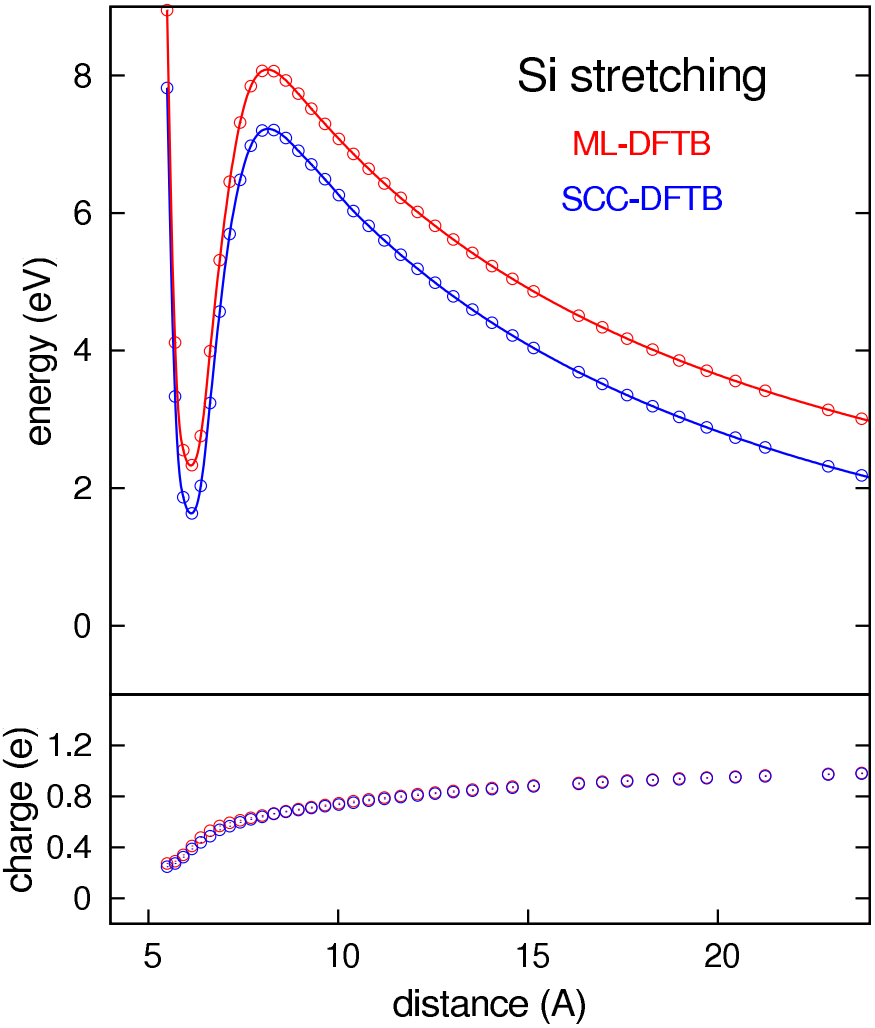

Figure 8: Energy and charge (respectively upper and lower panel) along a dissociation path for Si atom stretching. The abscissa is the distance between the stretched atom and the centre of mass of the cluster.

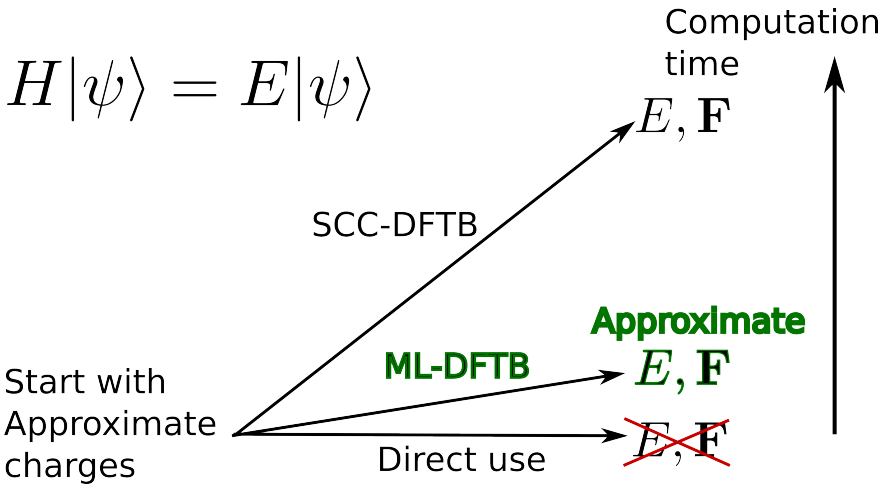

Figure 9: For Table of Contents Only

Supporting information

DFTB simulation of charged clusters using machine learning charge inference

Paul Guibourg, Léo Dontot, Pierre-Matthieu Anglade,* and Benoit Gervais

Laboratoire Cimap, UMR6252 — Université de Normandie Caen, École supérieure d'ingénieures de Caen, Commissariat à l'énergie atomique, Centre national de la recherche scientifique — 6 Boulevard du Maréchal Juin, 14050 Caen Cedex, France

E-mail: Pierre-Matthieu.Anglade@unicaen.fr

Phone: +33 (0)2 31452665

1 First and second order energy derivatives

To improve a charge guess, or a density matrix guess, we need an algorithm using explicitly the first and second order derivative of the energy, so that the direction of displacement and its magnitude can be obtained by means of a Newton-type algorithm. In the following, we detail the derivation of such an algorithm for DFTB. We start from the standard definition of the SCC-DFTB energy[1–3]:

$$E = \sum_{\mu,\nu} \rho_{\mu\nu} h_{\mu\nu} + \frac{1}{2} \sum_{A,B} q_A \gamma_{AB} q_B, \tag{1}$$

with the density matrix ρ defined from the molecular orbital coefficients $c_{n\mu}$ and weight ω_n of the orbital n as:

$$\rho_{\mu\nu} = \sum_n \omega_n c_{n\mu} c_{n\nu}, \tag{2}$$

and the net atomic charge on atomic centre A, q_A, is defined with the Mulliken definition as:

$$q_A = \sum_{\mu,\nu} \rho_{\mu\nu} S_{A,\mu\nu}. \tag{3}$$

Here, $S_{A,\mu\nu}$ denote the partial overlap matrix for atom A, and the complete overlap matrix is hereafter conventionally denoted as $S_{\mu\nu}$.

If we know an initial guess of charge for each atom A, we can generate a first set of molecular orbital coefficients $c^0_{n\mu}$ as the solution of the generalized eigenvalue problem:

$$\sum_\nu \left(h_{\mu\nu} + \sum_{A,B} q_A \gamma_{AB} S_{B,\mu,\nu} \right) c_{n\nu} = \sum_\nu S_{\mu\nu} c_{n\nu}. \tag{4}$$

We now seek for the rotation matrix e^R of orbital coefficients, which minimizes the energy E. Any antisymmetric matrix R is *a priori* acceptable. The coefficients $c_{n\mu}$ are defined from the initial coefficients $c^0_{n\mu}$ as:

$$c_{n\mu} = \sum_p (e^R)_{np} c^0_{p\mu}. \tag{5}$$

The rotation parameters that mix two orbitals with the same weight ω_n do not change the density matrix. They are thus discarded from the optimization process. Moreover, $R_{pq} = -R_{qp}$, and the total number of rotation parameters is of the order of half the product of occupied orbitals by the number of virtual or partially occupied orbitals.

We shall expand the energy with respect to R up to second order. We need the second order expansion of the molecular orbital coefficients:

$$c_{n\mu} = c^0_{n\mu} + \sum_p R_{np} c^0_{p\mu} + \frac{1}{2} \sum_{p,q} R_{np} R_{pq} c^0_{q\mu} \tag{6}$$

Using this expression, the band energy H decomposes in ascending order into $H = H_0 + H_1 + H_2$, with:

$$H_0 = \sum_{n,\mu,\nu} \omega_n c^0_{n\mu} c^0_{n\nu} h_{\mu\nu} \tag{7}$$

$$H_1 = \sum_{n,\mu,\nu} \omega_n \left(c^0_{n\mu} \sum_p R_{np} c^0_{p\nu} + c^0_{n\nu} \sum_p R_{np} c^0_{p\mu} \right) h_{\mu\nu} \tag{8}$$

$$H_2 = \sum_{n,\mu,\nu} \omega_n \left(\frac{1}{2} c^0_{n\mu} \sum_{pq} R_{np} R_{pq} c^0_{q\nu} + \sum_{pq} R_{np} R_{nq} c^0_{p\mu} c^0_{q\nu} + \frac{1}{2} c^0_{n\nu} \sum_{pq} R_{np} R_{pq} c^0_{q\mu} \right) h_{\mu\nu}. \tag{9}$$

We define $h_{nq} = h_{qn} = \sum_{\mu,\nu} c^0_{n\mu} c^0_{q\nu} h_{\mu\nu}$ to get:

$$H_0 = \sum_n \omega_n h_{nn} \tag{10}$$

$$H_1 = 2 \sum_n \omega_n \sum_p R_{np} h_{np} \tag{11}$$

$$H_2 = \sum_n \omega_n \sum_{pq} R_{np} R_{pq} h_{nq} + \sum_{pq} R_{np} R_{nq} h_{pq}. \tag{12}$$

We obtain a series of similar expressions for the expansion of q_A by substituting h_{nq} for W^A_{nq}. We define the Coulomb energy for a given pair of atom A, B as $G_{AB} = q_A \gamma_{AB} q_B$. The second order expansion of G_{AB} with respect to the matrix element R_{pq} is straightforward from the second order expansion of q_A.

1.1 First order derivatives

The first order derivative of H_1 with respect to R_{ax} reads:

$$\frac{\partial H_1}{\partial R_{ax}} = \sum_{n,p} \omega_n h_{np} \frac{\partial R_{np}}{\partial R_{ax}}. \tag{13}$$

Using $R_{np} = -R_{pn}$, the derivatives of R_{np} reads:

$$\frac{\partial R_{np}}{\partial R_{ax}} = \delta_{na}\delta_{px} - \delta_{pa}\delta_{nx}, \tag{14}$$

and since the matrix h is symmetric, we obtain:

$$\begin{aligned} \frac{\partial H_1}{\partial R_{ax}} &= 2\sum_p \omega_a h_{ap}\delta_{px} - \omega_x h_{xp}\delta_{pa} && (15) \\ &= 2(\omega_a - \omega_x)h_{ax}. && (16) \end{aligned}$$

When both orbitals a and x have the same weight, for example when they are both occupied or both empty, this expression vanishes.

There is a similar expression for the first order derivative of q_A with respect to R_{ax}. We use it to obtain the first order derivative of the Coulomb energy G_1:

$$\begin{aligned} \frac{\partial G_1}{\partial R_{ax}} &= \frac{1}{2}\sum_{AB} 2(\omega_a - \omega_x)S_{A,ax}\gamma_{AB}q_B + 2(\omega_a - \omega_x)S_{B,ax}\gamma_{AB}q_A && (17) \\ &= (\omega_a - \omega_x)\sum_{AB} \gamma_{AB}(q_B S_{A,ax} + q_A S_{B,ax}) && (18) \\ &= 2(\omega_a - \omega_x)\sum_{AB} \gamma_{AB} q_B S_{A,ax} && (19) \end{aligned}$$

1.2 Second order derivatives

We proceed in the same way to get the second order derivatives with respect to R_{ax} and R_{by}.

$$
\begin{aligned}
\frac{\partial^2 H_2}{\partial R_{ax}\partial R_{by}} &= \sum_{n,p,q} \omega_n h_{nq} \left(\frac{\partial R_{np}}{\partial R_{by}} \frac{\partial R_{pq}}{\partial R_{ax}} + \frac{\partial R_{np}}{\partial R_{ax}} \frac{\partial R_{pq}}{\partial R_{by}} \right) + \sum_{n,p,q} \omega_n h_{pq} \left(\frac{\partial R_{np}}{\partial R_{by}} \frac{\partial R_{nq}}{\partial R_{ax}} + \frac{\partial R_{np}}{\partial R_{ax}} \frac{\partial R_{nq}}{\partial R_{by}} \right) \\
&= \sum_{n,p,q} (\delta_{nb}\delta_{py} - \delta_{pb}\delta_{ny})(\delta_{pa}\delta_{qx} - \delta_{px}\delta_{qa})\omega_n h_{nq} \\
&+ \sum_{n,p,q} (\delta_{na}\delta_{px} - \delta_{pa}\delta_{nx})(\delta_{pb}\delta_{qy} - \delta_{qb}\delta_{py})\omega_n h_{nq} \\
&+ \sum_{n,p,q} (\delta_{nb}\delta_{py} - \delta_{pb}\delta_{ny})(\delta_{na}\delta_{qx} - \delta_{qa}\delta_{nx})\omega_n h_{pq} \\
&+ \sum_{n,p,q} (\delta_{na}\delta_{px} - \delta_{pa}\delta_{nx})(\delta_{nb}\delta_{qy} - \delta_{qb}\delta_{ny})\omega_n h_{pq}
\end{aligned} \tag{20}
$$

Each sum generates only 4 terms and we obtain a structure characteristic of 1-body operator. After reorganization of the different terms, the second order derivatives reads:

$$
\begin{aligned}
\frac{\partial^2 H_2}{\partial R_{ax}\partial R_{by}} &= \delta_{ab} h_{xy}(\omega_a - \omega_x + \omega_b - \omega_y) \\
&+ \delta_{ay} h_{by}(-\omega_a + \omega_x + \omega_b - \omega_y) \\
&+ \delta_{bx} h_{ay}(\omega_a - \omega_x - \omega_b + \omega_y) \\
&+ \delta_{xy} h_{ab}(-\omega_a + \omega_x - \omega_b + \omega_y)
\end{aligned} \tag{21}
$$

We obtain a similar expression for the derivatives of the atomic charge q_A at second order, by substituting h for W^A in the above expression.

The second order derivatives of the Coulomb energy G reads:

$$
\begin{aligned}
\frac{\partial^2 G_2}{\partial R_{ax}\partial R_{by}} &= \frac{1}{2} \sum_{AB} \left(\frac{\partial^2 q_A}{\partial R_{ax}\partial R_{by}} \gamma_{AB} q_B + q_A \gamma_{AB} \frac{\partial^2 q_B}{\partial R_{ax}\partial R_{by}} \right) \\
&+ \frac{1}{2} \sum_{AB} \left(\frac{\partial q_A}{\partial R_{ax}} \gamma_{AB} \frac{\partial q_B}{\partial R_{by}} + \frac{\partial q_A}{\partial R_{by}} \gamma_{AB} \frac{\partial q_B}{\partial R_{ax}} \right)
\end{aligned} \tag{22}
$$

To simplify the notation, we introduce the quantities:

$$K_{xy} = \frac{1}{2} \sum_{AB} \left(S_{A,xy} \gamma_{AB} q_B + q_A \gamma_{AB} S_{B,xy} \right) \tag{23}$$

$$\begin{aligned} L_{ax,by} &= \frac{1}{2} \sum_{AB} \left(S_{A,ax} \gamma_{AB} S_{B,by} + S_{A,by} \gamma_{AB} S_{B,ax} \right) \\ &= \sum_{AB} S_{A,ax} \gamma_{AB} S_{B,by} \end{aligned} \tag{24}$$

We finally express the second order derivatives of the Coulomb energy G as:

$$\begin{aligned} \frac{\partial^2 G_2}{\partial R_{ax} \partial R_{by}} &= \delta_{ab} K_{xy} (\omega_a - \omega_x + \omega_b - \omega_y) \\ &+ \delta_{ay} K_{by} (-\omega_a + \omega_x + \omega_b - \omega_y) \\ &+ \delta_{bx} K_{ay} (\omega_a - \omega_x - \omega_b + \omega_y) \\ &+ \delta_{xy} K_{ab} (-\omega_a + \omega_x - \omega_b + \omega_y) \\ &+ 4(\omega_a - \omega_x)(\omega_b - \omega_y) L_{ax,by} \end{aligned} \tag{25}$$

The second order derivative of the Coulomb energy is thus quite simple to evaluate once the summations have been performed to obtain K_{xy} and $L_{ax,by}$. In contrast to standard SCF calculations, the computation of the second order derivatives is usually not the bottleneck of the algorithm. Moreover, the matrices SA, ax are made of the elements of the DFTB overlap matrix whatever the index A and the memory requirement is limited.

2 ACSF parameters

We remind below the expression of the ACSF[4] used to described the atom environment and our choice of parameters taken from Weinreich *et al.*[5]. In the following formulas $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$.

$$g_{1,i} = \sum_j f_c\left(r_{ij}\right) \tag{26}$$

$$g_{2,i} = \sum_j f_c\left(r_{ij}\right) e^{-\eta(r_{ij}-r_s)^2} \tag{27}$$

$$g_{3,i} = \sum_j f_c\left(r_{ij}\right) \cos\left(\kappa r_{ij}\right) \tag{28}$$

$$g_{4,i} = 2^{1-\zeta} \sum_{j,k} \left(1 + \lambda \frac{\mathbf{r}_{ij} \cdot \mathbf{r}_{ik}}{r_{ij} r_{ik}}\right)^{\zeta} e^{-\eta\left(r_{ij}^2 + r_{ik}^2 + r_{jk}^2\right)} f_c\left(r_{ij}\right) f_c\left(r_{ik}\right) f_c\left(r_{jk}\right) \tag{29}$$

In the above expressions, the cutoff function f_c is defined as:

$$f_c(r) = \frac{1}{2}\left(\cos\frac{\pi r}{r_c} + 1\right) \tag{30}$$

with the cutoff parameter is $r_c = 6$ Å.

The corresponding parameters we used for g_2, g_3 and g_4 are listed in table 1, 2 and 3, respectively. The parameter values were simply taken from the work of Behler[4], without any optimization. For two chemical species, Si and C here, the total number of parameters is 120.

Table 1: Parameters of g_2 functions.

n°	$\eta\ [a_0^{-1}]$	$r_s\ [a_0]$
1	0.400	0.000
2	0.100	0.000
3	0.050	0.000
4	0.020	0.000
5	0.001	0.000
6	0.050	1.000
7	0.100	1.000
8	0.050	2.000
9	0.100	2.000

Table 2: Parameters of g_3 functions.

n°	$\kappa\ [a_0^{-1}]$
1	0.050
2	0.200
3	0.500
4	1.000
5	2.000

Table 3: Parameters of g_4 functions. Each line provides two functions, corresponding to $\lambda = +1$ and $\lambda = -1$.

n°	$\eta\ [a_0^{-1}]$	λ	ζ
1, 2	0.003	± 1	1
3, 4	0.008	± 1	1
5, 6	0.020	± 1	1
7, 8	0.050	± 1	1
9, 10	0.100	± 1	1
11, 12	0.003	± 1	2
13, 14	0.008	± 1	2
15, 16	0.020	± 1	2
17, 18	0.050	± 1	2
19, 20	0.100	± 1	2
21, 22	0.003	± 1	8
23, 24	0.008	± 1	8
25, 26	0.020	± 1	8
27, 28	0.050	± 1	8
29, 30	0.100	± 1	8

3 Timings

Because ML-DFTB offers the possibility to use directly approximate atomic charges, one of its compelling advantage is to bypass the SCC procedure, and thus to improve tremendously computation times. In the following we display some preliminary measurements.

As stated earlier, the ML-DFTB method presented is at the state of proof of concept. Our code works the following way : Our Python code — using ASE framework[6] — is launched, reads the atomic positions, uses them to predict the charges, and writes them as an input file to DFTB+. The later is then run and instead of performing a SCC calculation performs a single hamiltonian diagonalization, then exports eigenvalues, eigenvectors, all the matrix and matrix derivatives implied in the calculation into files. Our code reads those files, and compute the energy and forces of the ML-DFTB model. In term of timings, this implementation is clearly sub-optimal. Meanwhile improvements are already quite significant.

The extra time for ML charge evaluation through the NN is of the order of 15% of the orbital calculation including both Hamiltonian matrix computation and diagonalization. The computational times of one diagonalization for ML and SCC-DFTB are comparable; they also evolve similarly with system size. However the ML-DFTB requires a single diagonalization whatever the system.

Performing single processor computation on the same machine, we compute the timings shown in figure 1 (a) for DFT, SCC-DFTB, and ML-DFT. Since ML-DFTB bypasses SCC iterations, the present time improvement is already quite significant (about a factor of 10).

Figure 1 (b) underlines more significantly the expected benefits of the ML-DFTB in term of SCC cycles gain. It shows the behavior observed in our computations with DFTB+[7] while producing the reference calculations to parametrize the ML-DFTB method. As semiconductor SiC clusters exhibit typical behavior with respect to SCC convergence in DFTB and SCF convergence in DFT. As expected, convergence is fairly easy for the smallest systems and the number of iterations grows linearly with system size. For systems where SCC convergence is more difficult than SiC, for instance most metals, the expected benefit of a

ML-DFTB approach increases.

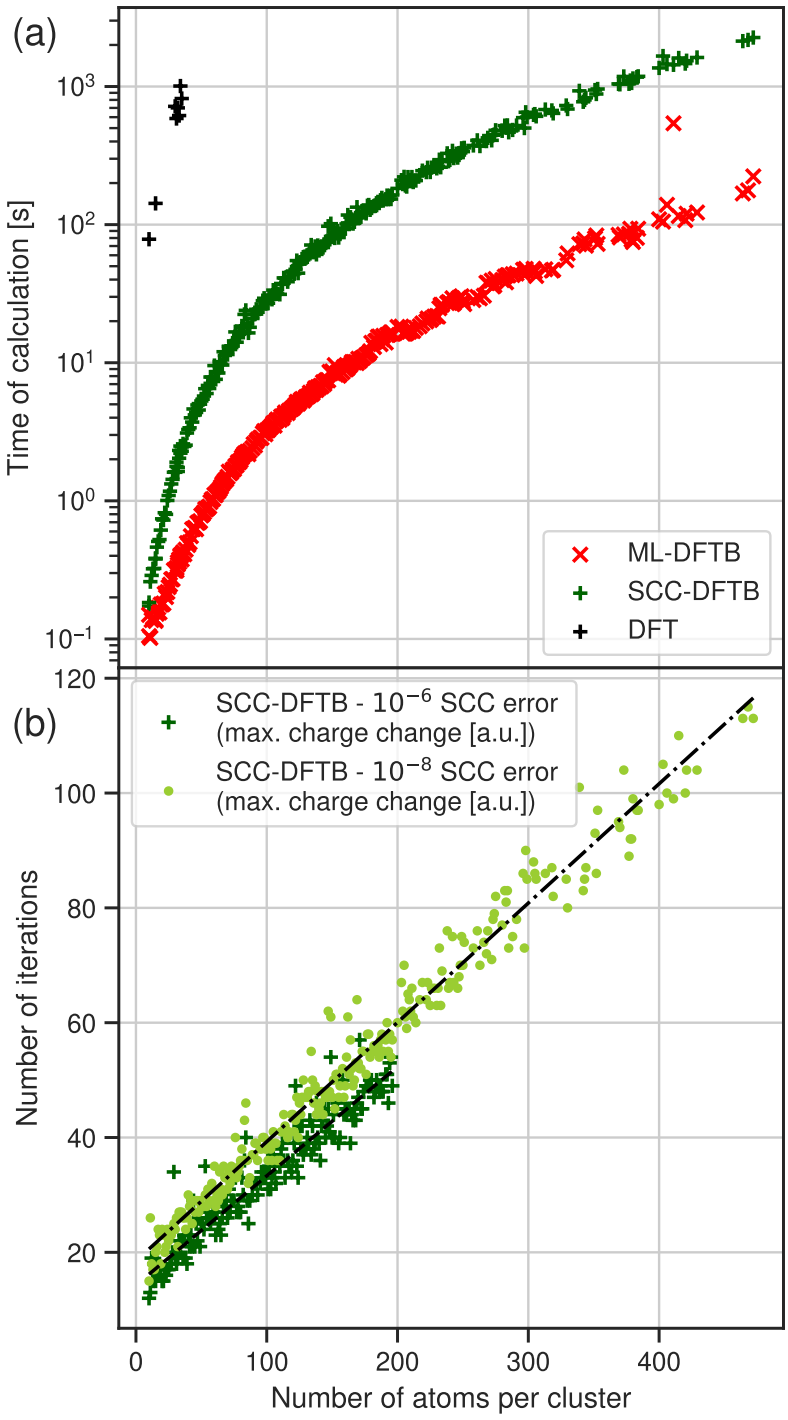

Figure 1: Computational informations for a few cluster used to test our ML model. (a) Actual computation timings; the DFT timings are done with BigDFT[8], the SCC-DFTB with DFTB+, and the ML-DFTB with our prototypical implementation of ML-DFTB. (b) Number of self-consistent charge cycles required to reach convergency for the same set of clusters.

References

(1) Elstner, M.; Porezag, D.; Jungnickel, G.; Elsner, J.; Haugk, M.; Frauenheim, T.; Suhai, S.; Seifert, G. Self-consistent-charge density-functional tight-binding method for simulations of complex materials properties. *Physical Review B* **1998**, *58*, 7260–7268.

(2) Pekka, K.; Ville, M. Density-Functional Tight-Binding for Beginners. *Computational Materials Science* **2009**, *47*, 237–253.

(3) Spiegelman, F.; Tarrat, N.; Cuny, J.; Dontot, L.; Posenitskiy, E.; Marti, C.; Simon, A.; Rapacioli, M. Density-functional tight-binding: basic concepts and applications to molecules and clusters. *Advances in Physics: X* **2019**, *5*, 1710252.

(4) Behler, J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *The Journal of Chemical Physics* **2011**, *134*, 074106.

(5) Weinreich, J.; Römer, A.; Paleico, M. L.; Behler, J. Properties of α-Brass Nanoparticles. 1. Neural Network Potential Energy Surface. *The Journal of Physical Chemistry C* **2020**, *124*, 12682–12695.

(6) Ask Hjorth Larsen, A. H.; Mortensen, J. J.; Blomqvist, J.; Castelli, I. E.; Christensen, R.; Dułak, M.; Friis, J.; Groves, M. N.; Hammer, B.; Hargus, C.; Hermes, E. D.; Jennings, P. C.; Jensen, P. B.; Kermode, J.; Kitchin, J. R.; Kolsbjerg, E. L.; Kubal, J.; Kaasbjerg, K.; Lysgaard, S.; Maronsson, J. B.; Maxson, T.; Olsen, T.; Pastewka, L.; Peterson, A.; Rostgaard, C.; Schiøtz, J.; Schütt, O.; Strange, M.; Thygesen, K. S.; Vegge, T.; Vilhelmsen, L.; Walter, M.; Zeng, Z.; Jacobsen, K. W. The atomic simulation environment—a Python library for working with atoms. *Journal of Physics: Condensed Matter* **2017**, *29*, 273002.

(7) Hourahine, B.; Aradi, B.; Blum, V.; Bonafé, F.; Buccheri, A.; Camacho, C.; Cevallos, C.; Deshaye, M. Y.; Dumitrică, T.; Dominguez, A.; Ehlert, S.; Elstner, M.; van der

Heide, T.; Hermann, J.; Irle, S.; Kranz, J. J.; Köhler, C.; Kowalczyk, T.; Kubař, T.; Lee, I. S.; Lutsker, V.; Maurer, R. J.; Min, S. K.; Mitchell, I.; Negre, C.; Niehaus, T. A.; Niklasson, A. M. N.; Page, A. J.; Pecchia, A.; Penazzi, G.; Persson, M. P.; Řezáč, J.; Sánchez, C. G.; Sternberg, M.; Stöhr, M.; Stuckenberg, F.; Tkatchenko, A.; Yu, V. W.-z.; Frauenheim, T. DFTB+, a software package for efficient approximate density functional theory based atomistic simulations. *The Journal of Chemical Physics* **2020**, *152*, 124101.

(8) Mohr, S.; Ratcliff, L. E.; Boulanger, P.; Genovese, L.; Caliste, D.; Deutsch, T.; Goedecker, S. Daubechies wavelets for linear scaling density functional theory. *The Journal of Chemical Physics* **2014**, *140*, 204110.