Maharavo Randrianarivony

On DFT molecular simulation for non-adaptive kernel approximation

PREPRINT

ON DFT MOLECULAR SIMULATION FOR NON-ADAPTIVE KERNEL APPROXIMATION

MAHARAVO RANDRIANARIVONY

Abstract. Direct computations of quantum entities can be very computationally expensive depending on the size of the configuration and the desired precision. We present here some preliminary results pertaining to stochastic methods for alleviating the numerical expense of quantum estimations. The initial information about the quantum energy originates from the Density Functional Theory. The determination of the parameters is performed by using methods stemming from machine learning. We survey the covariance method using log-likelihood for the statistical simulation. More emphasis is put at the position of equilibrium where the total atomic energy attains its minimum. In that process, the originally intensive data can be reproduced efficiently without loosing accuracy. For the numerical comparison, we use composites of germanium and silicon forming periodic structures. The amounts of germanium and silicon inside the composite Ge_xSi_{1-x} can be controlled. We examine the duration of the preparation overhead and we analyze the time gain by using our approach as compared to the direct DFT method.

1. Introduction

Nanotechnology tends to be the technology of focus in this century [4, 12]. Many of the macroscopically visible continuum applications seem to have attained nearly optimal state in computer implementation as well as in theoretical analysis. Not only the analytical and numerical treatments of nanotechnology are interesting but it has real-world applications in various disciplines including aircraft, automobile, electronic and medical engineering. In this paper, we would like to concentrate on the efficient treatment of quantum information. We present a method which is efficient for noisy data that contain measurement imperfection. Usually quantum data measurements are not completely exact for several reasons. First, treating the initial electronic Schrödinger equation is so computationally intensive that carrying it out without simplifications is generally computationally intractable. Second, the resolution used for discretizing the underlying differential equations might lead to inaccuracy if it is not sufficiently fine. Third, the simplified differential equations often contain parameters obtained from experimental measurements. Among other methods, DFT (Density Functional Theory) contains some parameters which are obtained from fitting procedures.

Key words and phrases. DFT, Tensor transformation, Kernel approximation.

The method presented here is based upon *machine learning* which uses the concept of *learning on the fly*. It means in general that when an atom or a configuration passes in one state, a certain computation is performed. Upon passing on that state for the next times, the computation is not repeated any more but experiences from previous computation is used. The machine learning process generates a structure which enables the efficient and fast access of such a methodology. Our motivation is to generate a system which is both accurate and fairly inexpensive to evaluate. Our method applies to all sorts of quantum information but we focus on the atomic energy in this paper. The availability of efficient and accurate evaluations can be used for subsequent applications. For example, that can be incorporated in a subsequent molecular dynamics. The evaluation of the resulting statistical data can provide derivatives and even second order derivatives.

The region where the system admits its equilibrium is examined more carefully than the remaining region during the approximation. That is, a geometry optimization must be performed initially in order to determine the position where the energy attains its minimum. The most general setup enables an estimation of an unknown function in arbitrary dimension such that noisy sample functions together with their higher derivatives are provided. In this document, we treat the simplified version where only the sample values are provided. Depending on the desired accuracy and the size of the atomic system, the determination of the energy can last several minutes by using direct DFT computations. The molecular configurations consist of germanium, silicon or composites of them using a few space group symmetry. The function to be reconstructed is the energy surface of a configuration when some tensor transformations are applied to DFT. The possible transformations include isotropic stretching, anisotropic stretching and Voigt tensor. Our numerical experiments reveal that the proposed statistical prefitting process seems to be a reasonable approach to obtain a good accuracy in an affordable computing time. The preparation overhead lasts very long but it needs only to be performed once for all and its output can be stored. The acceleration factor between the direct DFT and the kernel based approximation is approximately of order 10^5. Due to this significant acceleration gain, the preparation process shows to be worth calculating.

2. Theoretical methodology

2.1. **Ground state energy.** For a configuration of nuclei $\{\mathbf{a}_j\}$ of number N_u, the main problem in electronic structure computation involves the general Hamiltonian operator. We denote the coordinates of the i-th electron by $\mathbf{x}_i = (x_{i,1}, x_{i,2}, x_{i,3})$ and $\Delta_{\mathbf{x_i}} :=$

$\sum_{j=1}^{3} \partial^2/\partial x_{i,j}^2$. The stationary Hamiltonian reads

$$\mathcal{H} = -\sum_{n=1}^{N_u} \frac{\hbar^2}{2M_n}\Delta_{\mathbf{a}_n} \quad - \quad \frac{\hbar^2}{2m_e}\sum_{i=1}^{N_e}\Delta_{\mathbf{x}_i} - \sum_{i=1}^{N_e}\sum_{j=1}^{N_u}\frac{q_e Z_j}{\|\mathbf{x}_i - \mathbf{a}_j\|} + \frac{1}{2}\sum_{i=1}^{N_e}\sum_{j\neq i}^{N_e}\frac{q_e^2}{\|\mathbf{x}_i - \mathbf{x}_j\|} + \frac{1}{2}\sum_{n=1}^{N_u}\sum_{m\neq n}^{N_u}\frac{Z_n Z_m}{\|\mathbf{a}_n - \mathbf{a}_m\|} \tag{2.1}$$

where $\hbar$ is the Planck constant divided by 2π, M_n represents the mass of the n-th nucleus, q_e is the electronic charge, m_e the electron mass while the charge of the j-th nucleus is $Z_j = |q_e|n_j$ in which n_j is the atomic number. The kinetic energies of the nucleus and electron are

$$\Big\langle \Psi \Big| -\sum_{n=1}^{N_u}\frac{\hbar^2}{2M_n}\nabla_{\mathbf{a}_n}^2 \Big| \Psi \Big\rangle, \qquad \Big\langle \Psi \Big| -\frac{\hbar^2}{2m_e}\sum_{i=1}^{N_e}\nabla_{\mathbf{x}_i}^2 \Big| \Psi \Big\rangle \tag{2.2}$$

where the wave function Ψ solution to the eigenvalue problem

$$\mathcal{H}\Psi = E\Psi, \qquad \Psi = \Psi(\mathbf{a}_1, ..., \mathbf{a}_{N_a}, \mathbf{x}_1, ..., \mathbf{x}_{N_e}). \tag{2.3}$$

The third summations of (2.1) provides the nucleon-electron Coulomb attraction energy. The two last summations are the inter-electron and inter-nuclei repulsive energies.

For the *Born-Oppenheimer* or adiabatic approximation, one assumes that the mass and the volume of the atoms are very large in comparison to those of the electrons. Thus, the atoms move comparatively slower than the electrons. As a consequence, one treats the time-independent Hamiltonian operator with respect to the a set of nuclei $\mathbf{a}_i$ which are supposed to be stationary. That is, the electronic structure is governed by the expression

$$\mathcal{H}_{\mathbf{BO}} = -\frac{\hbar^2}{2m_e}\sum_{i=1}^{N_e}\Delta_{\mathbf{x}_i} + \sum_{i=1}^{N_e}\sum_{j=1}^{N_u}\frac{q_e Z_j}{\|\mathbf{x}_i - \mathbf{a}_j\|} + \frac{1}{2}\sum_{j=1}^{N_e}\sum_{i=1}^{j-1}\frac{q_e^2}{\|\mathbf{x}_i - \mathbf{x}_j\|}. \tag{2.4}$$

The atom-electron interaction and the inter-electron interaction operators $\mathcal{U}_{\mathbf{AE}}$ and $\mathcal{U}_{\mathbf{EE}}$ as follows

$$\mathcal{H}_{\mathbf{BO}} = \mathcal{K} + \mathcal{U}_{\mathbf{AE}} + \mathcal{U}_{\mathbf{EE}}, \qquad \text{where} \tag{2.5}$$

$$\mathcal{U}_{\mathbf{AE}}(\mathbf{x}_1, ..., \mathbf{x}_{N_e}) = \sum_{i=1}^{N_e} v_{\mathbf{AE}}(\mathbf{x}_i), \qquad \mathcal{U}_{\mathbf{EE}}(\mathbf{x}_1, ..., \mathbf{x}_{N_e}) = \frac{1}{2}\sum_{j=1}^{N_e}\sum_{i=1}^{j-1} v_{\mathbf{EE}}(\mathbf{x}_i, \mathbf{x}_j) \tag{2.6}$$

$$v_{\mathbf{AE}}(\mathbf{x}) := \sum_{j=1}^{N_u}\frac{1}{\|\mathbf{a}_j - \mathbf{x}\|}, \qquad v_{\mathbf{EE}}(\mathbf{x}, \mathbf{x}') := \frac{1}{\|\mathbf{x} - \mathbf{x}'\|}. \tag{2.7}$$

The domain of computation is $\mathbf{\Omega} \subset \mathbb{R}^3$ which is supposed sufficiently larger than the nuclei cloud $\{\mathbf{a}_j\}_{j=1}^{N_u}$ such that the above operator has neglecting influence beyond it. We

consider the domain of simulation By considering the electron spins which take values $+1/2$ or $-1/2$, we are searching for the electronic wave function

$$\Psi : \Big(\mathbf{\Omega} \times \{-1/2, +1/2\}\Big)^{N_e} \longrightarrow \mathbb{R} \qquad \text{verifying the eigenproblem}$$

$$\mathcal{H}_{\mathbf{BO}}\Psi(\gamma_1, ..., \gamma_{N_e}) = \lambda\Psi(\gamma_1, ..., \gamma_{N_e}) \quad \text{where} \quad \gamma_i = (\mathbf{x}_i, \sigma_i) \in \mathbf{\Omega} \times \{-1/2, +1/2\} \tag{2.8}$$

such that Ψ is antisymmetric $\Psi(..., \gamma_i, ..., \gamma_j, ...) = -\Psi(..., \gamma_j, ..., \gamma_i, ...)$ for all distinct $i, j = 1, ..., N$. Exact solutions for equation (2.8) are only known for very few special cases. Hence, numerical methods must be used for the general cases. The ground state energy corresponds to the smallest eigenvalue $\lambda_{\min}$ of (2.8). The antisymmetrization operator $\mathcal{A}$ applied to any N-variate function f is defined by

$$\mathcal{A}(f) := \frac{1}{N!} \sum_{P \in \mathbf{\Pi}_N} \operatorname{sgn}(P) f \circ P \tag{2.9}$$

in which $\mathbf{\Pi}_N$ is the set of permutations over $\{1, ..., N\}$ and $\operatorname{sgn}(P)$ designates the signature of a permutation P. If f is a tensor product function as $f = \otimes_{i=1}^{N} f_i$, then the antisymmetrizer coincides with the Slater determinant or wedge product

$$\mathcal{A}\big(\bigotimes_{i=1}^{N} f_i\big)(\gamma_1, ..., \gamma_N) = \frac{1}{N!} \begin{vmatrix} f_1(\gamma_1) & f_1(\gamma_2) & ... & f_1(\gamma_N) \\ f_2(\gamma_1) & f_2(\gamma_2) & ... & f_2(\gamma_N) \\ ... & ... & ... & ... \\ f_N(\gamma_1) & f_N(\gamma_2) & ... & f_N(\gamma_N) \end{vmatrix}. \tag{2.10}$$

The antisymmetrization operator $\mathcal{A}$ has the properties that it commutes with the Hamiltonian operator and that for an $N \times N$ matrix $\mathbf{M}$ such as $\mathcal{A}(\mathbf{M}\Phi) = \det(\mathbf{M})\mathcal{A}(\Phi)$. The Hartree-Fock approach is the variational formulation on the Hamiltonian operator (2.4) where the trial functions are antisymmetric functions. The main difficulty is that the problem is of $3N$-dimension without taking the electronic spins into account. In addition, on account of the antisymmetric property, a direct use of the Slater determinant often produces computations having orders of $\mathcal{O}(N!)$ which are very expensive. Counting the electronic spins lead usually to a factor of 2^N which makes the computation even more intractable.

2.2. **Quantum local simplification.** Some simplifications of the stationary Hamilton operators have already been proposed. For the DFT, one solves a set of equations for each electron. The similarity of the solutions is then derived from the theory of Kohn-Sham [5, 20] which consists in replacing the complicated initial problem into several ones. For each $i = 1, ..., N_e$

$$\big(-\frac{1}{2}\Delta + V_{\text{eff}}(\mathbf{x})\big)\psi_i(\mathbf{x}) = E_i\psi_i(\mathbf{x}) \tag{2.11}$$

where V_{eff} is the effective potential energy which depends implicitly on the total electron density $\rho(\mathbf{x}) = \sum_{i=1}^{N_e} |\psi_i(\mathbf{x})|^2$ such that $V_{\text{eff}}(\mathbf{x}) = V_{\text{eff}}[\rho(\mathbf{x})](\mathbf{x})$. The problem is then reduced from dimensions $3N_e$ to N_e sets of $3D$ smaller problems. The influence of one electron with respect to the other electron is measured by the total electron density. These approaches enable the treatment of Hamiltonian problem even for an electronic structure having a large number of particles on a single desktop. The effective potential is constituted of the Hartree potential V_H, the exchange correlation potential V_{XC} and the external electrostatic field such as

$$V_{\text{eff}}[\rho(\mathbf{x})](\mathbf{x}) = V_{\text{H}}[\rho(\mathbf{x})](\mathbf{x}) + V_{\text{XC}}[\rho(\mathbf{x})](\mathbf{x}) + V_{\text{ext}}[\rho(\mathbf{x})](\mathbf{x}) \tag{2.12}$$

in which the Hartree potential is the inverse of the Poisson operator such as $\Delta V_{\text{H}}(\text{x}) = -4\pi\rho(\mathbf{x})$. For its evaluation, either a Poisson problem is solved or one convolves with the Green fundamental solution such as $V_H(\mathbf{x}) = \int \rho(\widetilde{\mathbf{x}})/\|\mathbf{x}-\widetilde{\mathbf{x}}\| d\widetilde{\mathbf{x}}$. The main feature of DFT is that one has to approximate the potential by using some correction terms known as exchange-correlation potential [9, 10]. That is usually done by LDA (Local Density Approximation) or GGA (Generalized Gradient Approximation). Those expressions contain some parameters which are guessed or obtained from some experimental measurements in term of the Weigner-Seitz radius. Analytic expressions of the correlation energy are only known in a few special cases which mainly consist of the high and low density limits. The external electrostatic field potential $V_{\mathbf{ext}}$ is provided by the kernel $\sum_{i=1}^{N_u} z_i/\|\mathbf{x}-\mathbf{x}_i\|$. The above exchange-correlation potential is related to the exchange-correlation energy by $V_{\text{XC}} = \delta E_{XC}/\delta\rho$ where one expresses $E_{\text{XC}} = E_{\text{X}} + E_{\text{C}}$ as the exchange and the correlation parts. In term of the exchange-correlation energy density ε_{XC} one has

$$E_{\text{XC}}[\rho] = \int \varepsilon_{\text{XC}}[\rho](\mathbf{x})\rho(\mathbf{x})d\mathbf{x} \qquad \text{where} \qquad \varepsilon_{\text{XC}} = \varepsilon_{\text{X}} + \varepsilon_{\text{C}} \tag{2.13}$$

For the local density approximation (LDA), the exchange energy density is expressed as $\varepsilon_X^{LDA}(\rho) = -0.75(3\rho(\mathbf{x})/\pi)^{1/3}$ so that $E_{\text{X}}[\rho] = -0.75\big(\frac{3}{\pi}\big)^{1/3}\int \rho(\mathbf{x})^{4/3}d\mathbf{x}$. Analytic values of the correlation energy density are only known for some extreme cases. For the high density limit, the exchange correlation energy density as approximated by $\varepsilon_C = A\ln(r_s) + B + r_S(C\ln(r_S)) + D$ as the Weigner-Seitz radius r_s is very small. For the low density limit where r_s is very large, one has $\varepsilon_C = 0.5((g_0/r_s) + (g_1/r_s^{3/2}) + (g_2/r_s^{5/2}) + \cdots)$. For other values of r_s, some interpolation of those extreme values is considered. For example, by using the VWN-approximation (Vosko, Wilk, Nusair) as in [18], one has

$$\varepsilon_C(r_s,\zeta)^{\text{VWN}} = \varepsilon_C(r_s,0) + \varepsilon_a(r_s)\frac{f(\zeta)}{f''(0)}(1-\zeta^4) + \big[\varepsilon_C(r_s,1) - \varepsilon_C(r_s,0)\big]f(\zeta)\zeta^4 \tag{2.14}$$

where $f(\zeta) = 0.5((1+\zeta)^{4/3} + (1-\zeta)^{4/3} - 2)/(2^{1/3} - 1)$ while each one of $\varepsilon_C(r,0)$, $\varepsilon_C(r,1)$ and $\varepsilon_a(r)$ is of the form

$$\varepsilon_{C/a}(x) = \frac{A}{2}\Big(\ln\frac{x^2}{X(x)} + \frac{2b}{Q}\arctan\frac{Q}{2x+b}\Big) - \tag{2.15}$$

$$\frac{A}{2}\frac{bx_0}{X(x_0)}\Big(\ln\frac{(x-x_0)^2}{X(x)} + \frac{2(b+2x_0)}{Q}\arctan\frac{Q}{2x+b}\Big) \tag{2.16}$$

in which $x = \sqrt{r_s}$, $X(x) = x^2 + bx + c$ and $Q = \sqrt{4c - b^2}$. The constants A, x_0, b, c are fitting parameters which are different for $\varepsilon_C(r,0)$, $\varepsilon_C(r,1)$ and $\varepsilon_a(r)$. Once the solution E_i to (2.18) for all $i = 1, ..., N_e$ becomes known, the Khon-Sham approach uses the approximation to E of (2.1) by

$$E_{\mathbf{KS}} = \sum_{i=1}^{N_e} E_i - \frac{1}{2}\int \rho(\mathbf{x})V_H(\mathbf{x}) + \int \rho(\mathbf{x})\Big(E_{\mathrm{XC}}[\rho(\mathbf{x})] - V_{\mathrm{XC}}[\rho(\mathbf{x})]\Big)d\mathbf{x}. \tag{2.17}$$

The main improvement from LDA to GGA is that the exchange-correlation energy does not depend only on the total electron density but also on its gradient such as $E_{\mathrm{XC}}^{\mathrm{GGA}} = E_{\mathrm{XC}}^{\mathrm{GGA}}[\rho, \nabla\rho](\mathbf{x})$. Because the imperfections caused by estimation of parameters from experimental measurements, the DFT approach is not as accurate as the initial equations. But the statistical fitting approach proposed in this paper can deal with noisy data. The eigenvalue problem in (2.11) is nonlinear because its variational operator

$$\left\langle \mathbf{\Psi} \left| -\frac{1}{2}\nabla^2 + V_{\mathrm{eff}}[\rho^{(k)}](\mathbf{x}) \right| \mathbf{\Psi} \right\rangle, \tag{2.18}$$

depends on ρ which in turn depends on ψ_i. It is solved [20] by using a sequence of the linear eigenvalue problems SCF (Self Consistent Field). After assembling the operator (2.18), one solves the linear eigenvalue problem for the smallest eigenvalue $\lambda_{\min}^{(k)}$. One determines the new iterate $\psi_i^{(k)}(\mathbf{x})$ as the eigenfunction corresponding to $\lambda_{\min}^{(k)}$. In practice, one represents $\mathbf{\Psi}$ as a set of basis which are usually plane waves [19], Finite Element Method [16, 17], or wavelets [7, 13].

2.3. **Kernel-based approximation.** In this section, we will survey the main points about kernel-based approximation which are relevant in the quantum approximation. The most general setup of that approximation in any dimension D consists in accepting some inputs which are $\mathbf{x}_i \in \mathbb{R}^D$, $y_i \in \mathbb{R}$, $\mathbf{g}_i \in \mathbb{R}^D$ and $\mathbf{h}_i \in \mathbb{R}^{D\times D}$ such that

$$y_i = f(\mathbf{x}_i) + \varepsilon_i, \qquad \mathbf{x}_i \in \mathbb{R}^D, \qquad i \in \mathcal{J} \tag{2.19}$$

$$\mathbf{g}_i = \nabla f(\mathbf{x}_i) + \boldsymbol{\kappa}_i, \qquad i \in \mathcal{N} \tag{2.20}$$

$$\mathbf{h}_i = \left[\frac{\partial f}{\partial x_p \partial x_q}(\mathbf{x}_i)\right]_{p,q=1}^{D} + \boldsymbol{\lambda}_i, \qquad i \in \mathcal{M} \tag{2.21}$$

in which $\mathcal{N}$ and $\mathcal{M}$ are some subsets of $\mathcal{J}$ while ε_i, $\boldsymbol{\kappa}_i$, $\boldsymbol{\lambda}_i$ are measurement imperfections. In general, using the gradients is computationally very expensive so that the set $\mathcal{N}$ is much smaller than the set $\mathcal{J}$ while the Hessians are even more restrictively used. In this paper, we use only data measurements such that $\mathcal{N} = \emptyset$ and $\mathcal{M} = \emptyset$. The correlation [11] function to be used is the Matern function

$$k_{\text{MATERN}}(r) := \frac{2^{1-\nu}}{\Gamma(\nu)}\Big(\frac{\sqrt{2\nu}r}{\ell}\Big)^{\nu} K_\nu\Big(\frac{\sqrt{2\nu}r}{\ell}\Big) \tag{2.22}$$

where (ν, ℓ) are positive hyper-parameters and K_ν is the modified Bessel function. In dimension D, the spectral density of the Matern function is

$$S(s) = \frac{2^D \pi^{D/2}\Gamma(\nu + D/2)(2\nu)^\nu}{\Gamma(\nu)\ell^{2\nu}}\Big(\frac{2\nu}{\ell^2} + 4\pi^2 s^2\Big)^{(-\nu+D/2)}. \tag{2.23}$$

If ν tends to infinity, one obtains the squared exponential case. For the special case of half-integers such as $\nu = p + 1/2$, one obtains a product of an exponential and a polynomial of order p such that (2.22) becomes

$$k_\nu(r) = \exp\Big(-\frac{\sqrt{2\nu}r}{\ell}\Big)\frac{\Gamma(p+1)}{\Gamma(2p+1)}\sum_{i=0}^{p}\frac{(p+i)!}{i!(p-i)!}\Big(\frac{\sqrt{8\nu}r}{\ell}\Big)^{p-i}. \tag{2.24}$$

For the most applied cases where $\nu = 3/2$ and $\nu = 5/2$, one has

$$k_{3/2}(r) = \Big(1 + \frac{\sqrt{3}r}{\ell}\Big)\exp\Big(-\frac{\sqrt{3}r}{\ell}\Big), \tag{2.25}$$

$$k_{5/2}(r) = \Big(1 + \frac{\sqrt{5}r}{\ell} + \frac{5r^2}{3\ell^2}\Big)\exp\Big(-\frac{\sqrt{5}r}{\ell}\Big). \tag{2.26}$$

Suppose N is the number of observations such that $\text{Card}(\mathcal{J}) = N$ in (2.19). For noise-free observations, one has the covariance

$$\text{Cov}\big(f(\mathbf{x}_p), f(\mathbf{x}_q)\big) = K(\mathbf{x}_p, \mathbf{x}_q). \tag{2.27}$$

For a noisy data $y_i = f(\mathbf{x}_i) + \varepsilon$ where the additional noise follows the Gaussian distribution $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$, the covariance becomes

$$\text{Cov}(\mathbf{y}_p, \mathbf{y}_q) = K(\mathbf{x}_p, \mathbf{x}_q) + \sigma_n^2 \delta_{pq}. \tag{2.28}$$

Let us denote the sampling points by $X = [\mathbf{x}_1, ..., \mathbf{x}_N]^T$ which is a matrix having the size $N \times D$ where D is the dimension of the trial variables $\mathbf{x}_i$. The latent function and the set of observations are defined in a similar fashion such that $\mathbf{f} = [f_1, ..., f_N]^T$ and $\mathbf{y} = [y_1, ..., y_N]^T$. The marginal likelihood is the integral of the likelihood and the prior such as

$$p(\mathbf{y}|X) = \int p(\mathbf{y}|\mathbf{f})\, p(\mathbf{f}|X) d\mathbf{f}. \tag{2.29}$$

We denote by $K = K(X, X)$ the covariance matrix whose entries are $K(\mathbf{x}_p, \mathbf{x}_q) = k_{\text{MATERN}}(r)$ where r will be defined below as a generalized distance between $\mathbf{x}_p$ and $\mathbf{x}_q$. We

will use also $|K|$ to denote the determinant of K. If $\mathbf{f}|X$ follows the Gaussian distribution $\mathcal{N}(\mathbf{0}, K)$ and one has no noise, then the marginal likelihood is

$$p(\mathbf{f}|X) = (2\pi)^{-N/2}|K|^{-1/2}\exp\left[-\frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right]. \tag{2.30}$$

By taking the logarithm, the computation of the marginal likelihood is as follows

$$p(\mathbf{f}|X)|K|^{1/2}(2\pi)^{N/2} = \exp\left[-\frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f}\right] \tag{2.31}$$

$$\log\big[p(\mathbf{f}|X)\big] + \frac{1}{2}\log|K| + \frac{N}{2}\log(2\pi) = -\frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f} \tag{2.32}$$

from which one obtains the log marginal likelihood

$$\log\big[p(\mathbf{f}|X)\big] = -\frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f} - \frac{1}{2}\log|K| - \frac{N}{2}\log(2\pi). \tag{2.33}$$

In the presence of noise $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$ by using (2.28), the negative log marginal likelihood becomes

$$-\log\big[p(\mathbf{y}|X)\big] = \frac{1}{2}\mathbf{y}^T\big(K + \sigma_n^2 I_{N,N}\big)^{-1}\mathbf{y} + \frac{1}{2}\log\big|K + \sigma_n^2 I_{N,N}\big| + \frac{N}{2}\log(2\pi). \tag{2.34}$$

The first term $\frac{1}{2}\mathbf{y}^T\big(K + \sigma_n^2 I_{N,N}\big)^{-1}\mathbf{y}$ is the *data-fit term* while the second term $\frac{1}{2}\log\big|K + \sigma_n^2 I_{N,N}\big|$ and the last term $\frac{N}{2}\log(2\pi)$ are respectively the *complexity term* and the *normalization term*. For a set of test points $\mathbf{x}_i^* \in \mathbb{R}^D$, the expressions X^*, $K(X, X^*)$ and $K(X^*, X^*)$ are defined in a similar way as X and $K(X, X)$. The predicted estimation follows the Gaussian distribution

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma_n^2 I_{N,N} & K(X, X^*) \\ K(X^*, X) & K(X^*, X^*) \end{bmatrix}\right). \tag{2.35}$$

That is to say, for given X and $\mathbf{y}$, the predictive distribution for a covariate vector X^* is a Gaussian admitting the following mean and variance

$$E\big[\mathbf{f}^*|X^*, X\big] = K(X^*, X)\big[K(X,X) + \sigma_n^2 I_{N,N}\big]^{-1}\mathbf{y} \tag{2.36}$$

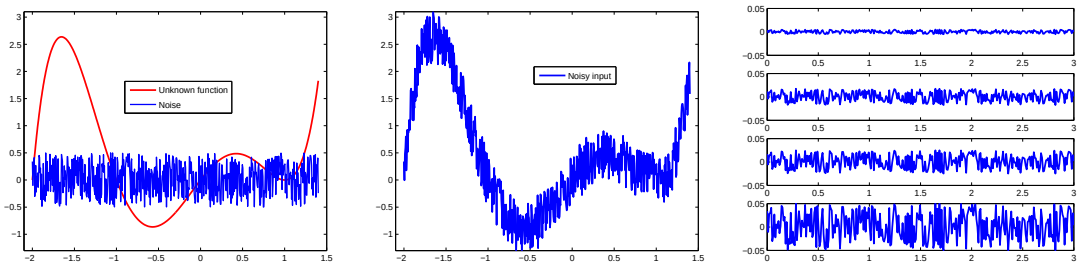
$$\mathrm{Var}\big[\mathbf{f}^*|X^*, \mathbf{y}, X\big] = K(X^*, X^*) - K(X^*, X)\big[K(X,X) + \sigma_n^2 I_{N,N}\big]^{-1}K(X, X^*) \tag{2.37}$$

where those expressions are dependent on some set of hyper-parameters which we discuss below.

The value of the generalized distance r for given $\mathbf{x}_p \in \mathbb{R}^D$ and $\mathbf{x}_q \in \mathbb{R}^D$ is

$$r^2 = (\mathbf{x}_p - \mathbf{x}_q)^T \begin{bmatrix} 1/\lambda_1^2 & & \\ & \ddots & \\ & & 1/\lambda_D^2 \end{bmatrix} (\mathbf{x}_p - \mathbf{x}_q) \tag{2.38}$$

in which λ_1,..., λ_D are related to hyper-parameters. In general, the set of hyperparameters include $\theta = \log(\ell)$ where ℓ is from (2.25) and (2.26) as well as $\theta = \log(\lambda_i)$ where $i = 1, ..., D$.



(a) (b) (c)

FIGURE 1. (a)Unknown function and noise (b)Noisy input (c)Noise level.

The main objective is to determine the hyperparameters by optimizing the marginal likelihood. In practice, using the log marginal likelihood is more efficient to implement. In order to accelerate the speed of the nonlinear optimization, we need the gradients of the functional (2.34) with respect to the hyperparameters. For a hyper-parameter θ, one has the partial derivative

$$\frac{\partial}{\partial\theta}K^{-1} = -K^{-1}\frac{\partial K}{\partial\theta}K^{-1} \quad \text{and} \quad \frac{\partial}{\partial\theta}\log|K| = \text{trace}\left[K^{-1}\frac{\partial K}{\partial\theta}\right]. \tag{2.39}$$

From the chain rule, one obtains

$$\partial_\theta\Big[k(r(\mathbf{x}_p,\mathbf{x}_q))\Big] = k'(r)\partial_\theta\Big[r(\mathbf{x}_p,\mathbf{x}_q)\Big], \quad \text{where} \quad r(\mathbf{x}_p,\mathbf{x}_q) = \sqrt{\sum_{i=1}^{D}\frac{1}{\lambda_i}(x_{p,i}-x_{q,i})^2} \tag{2.40}$$

This is singular when $\mathbf{x}_p = \mathbf{x}_q$ which occurs on the diagonal entries of the covariance matrix. In fact, one has

$$\partial_{\lambda_\mu} r(\mathbf{x}_p,\mathbf{x}_q) = -\frac{1}{r(\mathbf{x}_p,\mathbf{x}_q)}\left(\frac{x_{p,\mu}-x_{q,\mu}}{\lambda_\mu}\right)^2 \qquad \text{for} \quad \mu = 1,...,D. \tag{2.41}$$

Working with $R := r^2$ provides a regularization everywhere as $\partial_{\lambda_\mu}R(\mathbf{x}_p,\mathbf{x}_q)$ is regular. In addition, $\widetilde{k}(R) := k_{\text{MATERN}}(\sqrt{r})$ is smooth at the origin because one has indeed $k(R) = p_{\text{MATERN}}(\sqrt{2\nu}r/\ell)$ such that

$$p_{\text{MATERN}}(t) := f_{\text{MATERN}}(t)\exp(-t) \tag{2.42}$$

$$p'_{\text{MATERN}}(t) = \Big[f'_{\text{MATERN}}(t) - f_{\text{MATERN}}(t)\Big]\exp(-t) \tag{2.43}$$

where f_{MATERN} is a polynomial. In particular, for the cases where $\nu = 3/2$ and $\nu = 5/2$ in (2.25) and (2.26), one has $f_{3/2} = 1+t$ and $f_{5/2} = 1+t+(1/3)t^2$. In fact, one obtains the relation

$$\Big[f'_{\text{MATERN}}(t) - f_{\text{MATERN}}(t)\Big] = tq_{\text{MATERN}}(t) \tag{2.44}$$

where q_{MATERN} is a polynomial. Therefore, one deduces from (2.44)

$$\frac{d\widetilde{k}}{dR} = \frac{\sqrt{2\nu}}{2\ell\sqrt{R}}\left[f'_{\text{MATERN}}\Big(\frac{\sqrt{2\nu}\sqrt{R}}{\ell}\Big) - f_{\text{MATERN}}\Big(\frac{\sqrt{2\nu}\sqrt{R}}{\ell}\Big)\right]\exp\Big(-\frac{\sqrt{2\nu}\sqrt{R}}{\ell}\Big) \tag{2.45}$$

$$= \frac{\sqrt{2\nu}}{2\ell}q_{\text{MATERN}}\Big(\frac{\sqrt{2\nu}\sqrt{R}}{\ell}\Big)\exp\Big(-\frac{\sqrt{2\nu}\sqrt{R}}{\ell}\Big) \tag{2.46}$$

which is regular for all values of $r(\mathbf{x}_p, \mathbf{x}_q)$.

3. Simulation results

In this section, we would like to report on some results from computer simulation of the formerly described approach. First, we will present some results pertaining to general real valued multi-variate functions. That will be followed by some application in quantum simulation.

3.1. **Computer implementation.** The former theoretical approach was implemented by using C/C++, BLAS/LAPACK and NLOPT. The BLAS packet is used for the fast vector operations. We use LAPACK for the linear operations such as Cholesky factorization and dense matrix solvers. We use NLOPT for the nonlinear operations [6] for both the geometry optimization and the optimal hyper-parameters in the log marginal likelihood (2.34). NLOPT supports diverse nonlinear optimization operations [8] in which local optimizers are involved. A local one searches only inside a neighborhood of a certain provided starting initial guess. The optimizers are performed by using derivative free or gradient based algorithm which are available in NLOPT. Derivative free algorithms include BOBYQA (Bond Optimization BY Quadratic Approximation), COBYLA (Constrained Optimization BY Linear Approximation), NEWUOA (NEW Unconstrained Optimization Algorithm). Gradient-based methods include MMA (Method of Moving Asymptotes) and LBFGB (Limited memory Broyden-Fletcher-Goldfarb-Shanno). The code is a very developed version of the matlab implementation provided in [11]. The new additional enhancements from the matlab version consist of the following features. First, using NLOPT provides a lot of improvements as compared to the original matlab nonlinear conjugate gradients. That can be observed when both the number of points and the dimension become large in the nonlinear optimization of the hyper-parameters. In addition, we can also accept higher derivatives in the input apart from the functional values. One can use the entire gradient or only some components of it. Furthermore, the gradient of the kernel-based approximation can be evaluated. That can be done analytically instead of using finite difference. In addition, our C/C++ code admits some python interface enabling direct application to ATK which is the quantum package we use.

Dimension	Error in	Number of data				
		10	20	30	40	50
4	*Function*	1.4727e-01	2.4419e-02	2.3565e-02	1.6221e-02	1.4906e-02
	Gradient	4.7896e-01	1.1478e-01	1.0937e-01	8.6681e-02	7.7206e-02
5	*Function*	9.0570e-02	3.1160e-02	2.7457e-02	2.6331e-02	1.8296e-02
	Gradient	3.8708e-01	1.3668e-01	1.2931e-01	1.2187e-01	9.8637e-02
6	*Function*	1.0068e-01	3.5685e-02	2.7100e-02	2.0901e-02	1.8697e-02
	Gradient	3.8986e-01	1.5762e-01	1.2739e-01	1.1043e-01	1.0545e-01

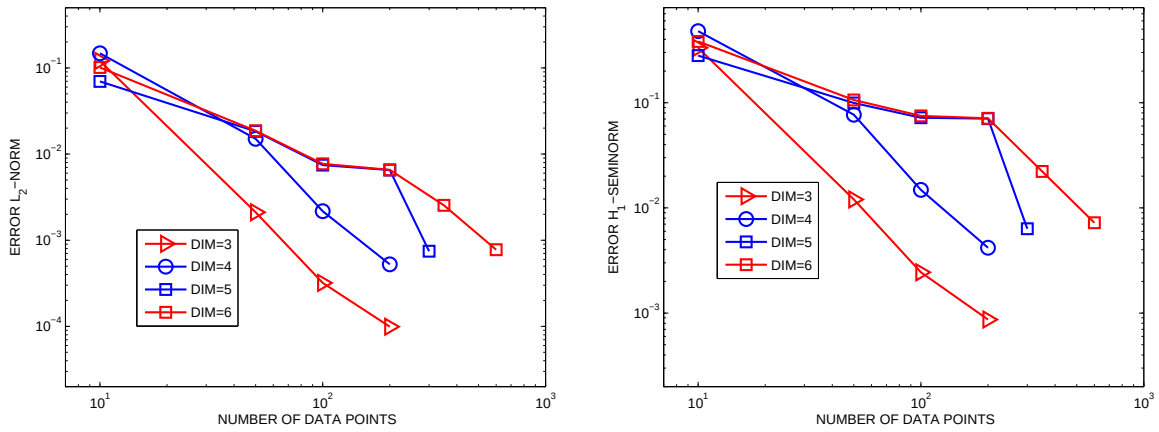
TABLE 3.1. Comparison w.r.t. $f(\mathbf{x}) = \sum_{i=1}^{D} \sin(x_i)$. Error of the function and gradient values on $[0, \pi]^D$ for dimension $D = 4, 5, 6$.

As a first test, we consider the reconstruction of the function $f(x_1, ..., x_D) = \sum_{i=1}^{D} \sin(x_i)$ by using the kernel-based approximation. Since that function does not present any special feature such as cusp or boundary layer or any special interesting region, we use only randomly generated points. The initial guess of the hyper-parameters is provided by the users. One can consider the determination of the final hyper-parameters as an unconstrained nonlinear optimization. On the other hand, we are sometimes interested in the local optimal variables which are next to the hyper-parameters $[\theta_1^0, ..., \theta_n^0]$ with which the data are generated as specified in Chapter 5, page 115 of [11]. In that case, for each hyper-parameter θ_i we consider the constraint interval $[\theta_i^0 - \delta, \theta_i^0 + \delta]$ where δ is some positive range. The appropriate value of $\delta > 0$ is for now specified by the user.

Dimension	# Data	Noise=0 Error	Noise=0.001 Error	Noise=0.01 Error	Noise=0.1 Error
4	50	2.7918e-02	2.7900e-02	2.7952e-02	5.9436e-02
	100	1.9100e-04	1.9030e-03	5.7590e-03	5.5989e-02
	200	1.1000e-04	5.5700e-04	5.3210e-03	5.2768e-02
	250	4.4000e-05	5.4000e-04	5.3220e-03	5.2619e-02
6	50	3.3249e-02	3.3255e-02	3.3562e-02	5.6176e-02
	100	5.6790e-03	5.6650e-03	7.3480e-03	5.4478e-02
	200	4.3530e-03	4.3630e-03	6.3610e-03	5.1767e-02
	250	3.9300e-04	6.7200e-04	4.9430e-03	5.1988e-02

TABLE 3.2. Influence of the noise. Average error on $[0, 1]^D$ for dimension $D = 4, 6$.

The results of the computations are collected in Table 3.1. The errors are computed for the function values as well as for the gradient values. The results exhibit the performance for different numbers of data for each dimension. In addition, we examine the effect of increasing the dimension D. In Fig. 2, we display the error plots in function of the number of data. We consider dimensions $D = 3, 4, 5, 6$ by analyzing the errors in the function evaluation and the gradient values which are respectively shown in Fig. 2(a) and Fig. 2(b) . We observe that the increase of the dimension does not really deteriorate the accuracy too much. That fact holds for both the function error and the gradient error. One needs certainly more points for higher dimensions but the increase of the number of additional points is not too significant.



(a) (b)

FIGURE 2. Increasing the dimensions: (a) Error in function values (b) Error in gradient values.

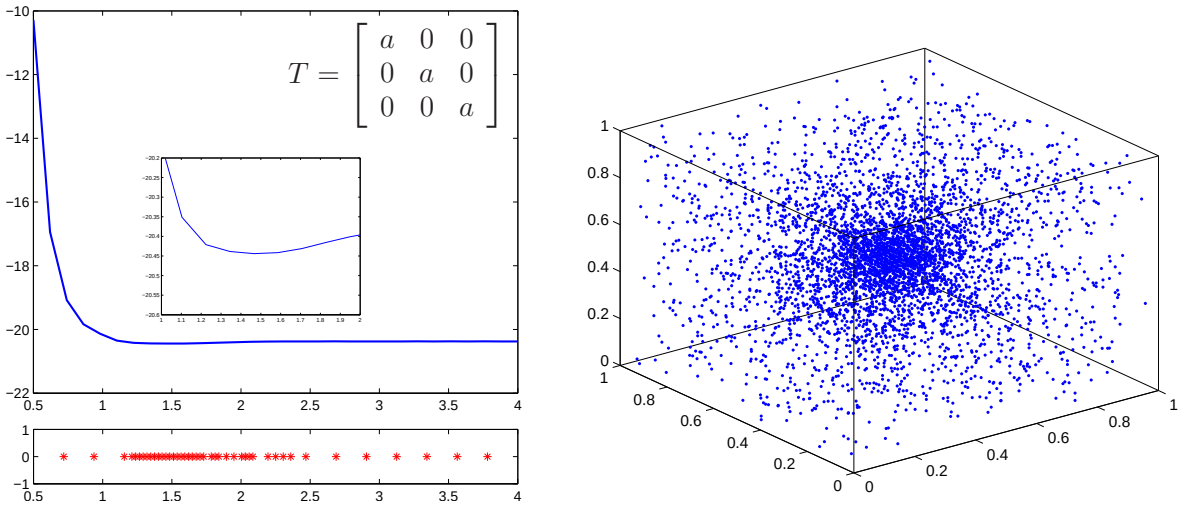
Since the real-world applications contain some noise in the provided data, we would like to examine now the influence of the noise on the reconstruction. We investigate only the case where the dimensions are $D = 4$ and $D = 6$ because the other dimensions have very similar behavior. One observes the change of the level of the noise in Figure 1(c) where the amplitude of the noise is very small on the top. We use randomly generated noise by using a factor of a unit random number generator and a given noise amplitude. The used data is the sum of the initial function and the appended imperfection. That is, one perturbs the exact function by additional noise as illustrated in Fig. 1(a) in order to obtain the final input such as in Fig. 1(b). The amplitude of the noise ranges from 0 to 0.1 which is an absolute value and not the noise signal ratio. The results are displayed on Table 3.2 where the errors are measured by using average pointwise gauge. A noise of amplitude 0.01 still permits the method to detect sinusoidal functions well. For excessively

large noise amplitudes, the function is completely buried under the noise and no accurate reconstruction is achieved.

3.2. **Unit cell transformation.** For the application to nano-simulation, we consider the unit cell generated by three vectors $\mathbf{u}_A$, $\mathbf{u}_B$, $\mathbf{u}_C \in \mathbb{R}^3$. After a transformation $T_{(\lambda_1,...,\lambda_D)}$, they become $\widetilde{\mathbf{u}}_A := T(\mathbf{u}_A)$, $\widetilde{\mathbf{u}}_B := T(\mathbf{u}_B)$, $\widetilde{\mathbf{u}}_C := T(\mathbf{u}_C) \in \mathbb{R}^3$. For a bulk configuration $\mathcal{K}$, we denote by $T(\mathcal{K})$ the transformed configuration where the fractional coordinates belonging to the transformed unit cell $T(\mathcal{K})$ remain the same as in the original configuration $\mathcal{K}$. Their values are within $[0, 1]$ before and after the transformations. That is, the fractional coordinates of the reference configuration $\mathcal{K}$ remain unchanged from beginning till the end of the computation. We are interested in the impact of the DFT energy by applying the transformation to the unit cell vectors. The transformation $T_{(\lambda_1,...,\lambda_D)}$ depends on D parameters so that the D-variate function to be approximated maps $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_D)$ to the energy such as

$$(\lambda_1, ..., \lambda_D) \longmapsto E = DFT\big[T_{(\lambda_1,...,\lambda_D)}(\mathcal{K})\big].$$

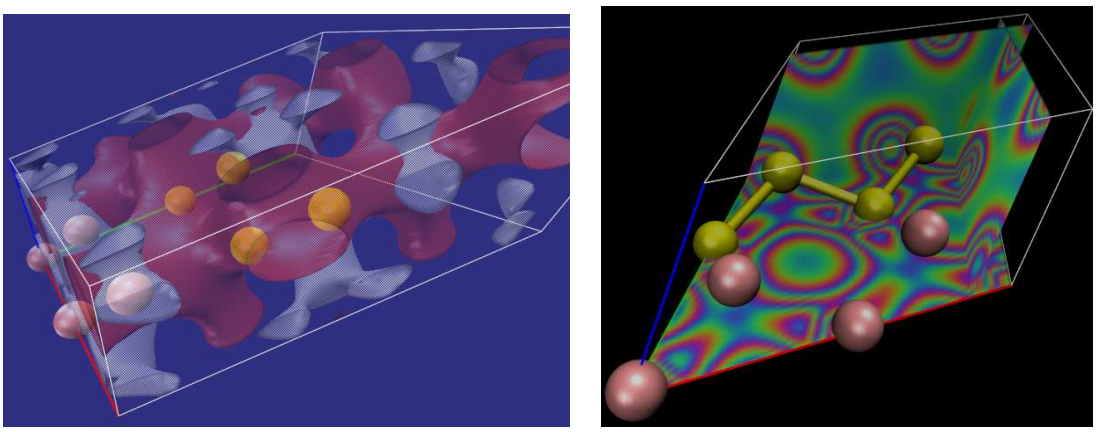
The first transformation consists of an isotropic one that corresponds to a unidimensional function which is a scaling. That transformation amounts to stretching and confining the unit cell equally in all x, y, z directions. Thus, the function to be approximated maps λ into $DFT\big[T_\lambda(\mathcal{K})\big]$ where one has $\widetilde{\mathbf{u}}_A := \lambda\mathbf{u}_A$, $\widetilde{\mathbf{u}}_B := \lambda\mathbf{u}_B$, $\widetilde{\mathbf{u}}_C := \lambda\mathbf{u}_C$.



(a) (b)

FIGURE 3. Accumulated sampling points at the geometric optimum: (a)Unidimensional, (b)Tridimensional.

The next transformations are anisotropic in the sense that the unit cell is scaled differently in the x, y, z directions. The first anisotropic transformation scales in a way that $\widetilde{\mathbf{u}}_A := \lambda_1\mathbf{u}_A$, $\widetilde{\mathbf{u}}_B := \lambda_2\mathbf{u}_B$, $\widetilde{\mathbf{u}}_C := \lambda_3\mathbf{u}_C$. That transformation corresponds to the case where the



(a) (b)

FIGURE 4. Electronic density for the DFT simulation of composite germanium/silicon.

dimension $D = 3$ and T can be represented as a diagonal matrix having $(\lambda_1, \lambda_2, \lambda_3)$ as entries. For the 2D anisotropic case, the transformation is given by

$$T_{(a,b)} = \begin{bmatrix} ab & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & b \end{bmatrix}. \tag{3.47}$$

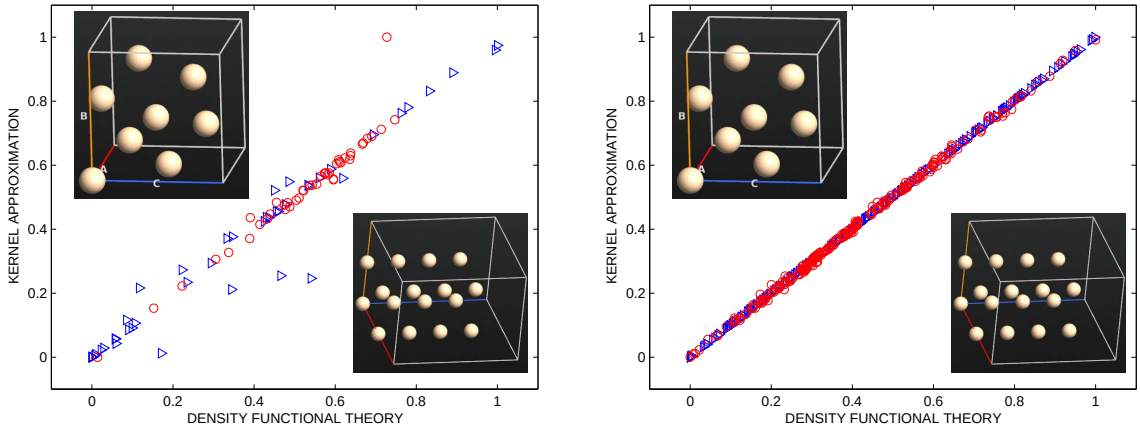
For the most general transformation, one has a strain tensor

$$T_{(\varepsilon_{xx},\varepsilon_{xy},\varepsilon_{xz},\varepsilon_{yy},\varepsilon_{yz},\varepsilon_{zz})} = \begin{bmatrix} 1+\varepsilon_{xx} & 0.5\varepsilon_{xy} & 0.5\varepsilon_{xz} \\ 0.5\varepsilon_{yx} & 1+\varepsilon_{yy} & 0.5\varepsilon_{yz} \\ 0.5\varepsilon_{zx} & 0.5\varepsilon_{zy} & 1+\varepsilon_{zz} \end{bmatrix}. \tag{3.48}$$

This is the Voigt strain tensor which is usually encountered in anisotropic transformations such as elasticity. This general transformation corresponds to a six-variate function since the Voigt strain matrix is supposed to be symmetric. The isotropic and 2D/3D anisotropic transformations are ensured to be nonsingular provided that the diagonal parameters are strictly positive. The tensor transformation in (3.48) is nonsingular as long as the parameters $[\boldsymbol{\varepsilon}]$ are not too large as the Voigt tensor approaches the identity transformation.

3.3. **The focus of the approximation.** Although we are interested in the general approximation of the quantum energy, we put more emphasis on the neighborhood of the position where the energy attains its minimal value because it defines the equilibrium of the system. The main objective is to obtain a good approximation next to the optimal point $(\lambda_1^{\text{opt}}, ..., \lambda_D^{\text{opt}})$ corresponding to

$$\text{argmin}_{(\lambda_1,...,\lambda_D)}\text{Energy}\left[T_{(\lambda_1,...,\lambda_D)}(\mathcal{K})\right] \tag{3.49}$$



(a) (b)

FIGURE 5. DFT vs. kernel approximation for two silicon configurations:(a) 45 sampling points (b) 225 sampling points.

for a molecular or bulk configuration $\mathcal{K}$. We have built a function that performs the geometry optimization using unpolarized single ζ DFT calculators for any given molecular configuration.

That function accepts as arguments the reference configuration, the DFT-calculator, the domain $[a, b]$ where the optimum is sought, the desired accuracy as well as the maximum number of evaluations of the nonlinear functional. In the case that a and b are floating point numbers, one searches for the optimum $\boldsymbol{\lambda}^{\text{opt}} = (\lambda_1^{\text{opt}}, ..., \lambda_D^{\text{opt}})$ within $[a, b]^D$. If they are arrays, each one of them must be of size D and the optimum $\boldsymbol{\lambda}^{\text{opt}}$ is sought within the hypercube $\prod_{i=1}^{D} \big[a[i-1], b[i-1]\big]$. Note that the accuracy is the tolerance in the precision of the variable $(\lambda_1^{\text{opt}}, ..., \lambda_D^{\text{opt}})$ and not the energy functional. This is the first instance in the program where the optimizer NLOPT is applied.

The Gaussian approximation are applied on a set of points $\boldsymbol{\lambda}_i = (\lambda_{1,i}, ..., \lambda_{D,i})$ together with their images $E = DFT\big[T_{(\lambda_{1,i},...,\lambda_{D,i})}(\mathcal{K})\big]$. In our applications, we accumulate very dense points in the neighborhood of the optimal value $\boldsymbol{\lambda}^{\text{opt}}$ while only few points are used elsewhere. An illustration of this situation is displayed in Fig. 3(a) for the unidimensional case where the energy curve is traced together with the samples accumulating at the equilibrium. A typical sampling point distribution for the spatial case is depicted in Fig. 3(b).

3.4. **Molecular simulation.** We give now some overview of the quantum simulator. The implementation of the quantum application was realized by using ATK (Atomistix ToolKit) [2, 3, 14] which consists of python scripts. The ATK has some GUI extension well known as VNL (Virtual NanoLab). For given nuclei coordinates $\{\mathbf{x}_1, ..., \mathbf{x}_{N_u}\}$, the

used quantum packet can provide the following energy and force

$$E_{\min}(\mathbf{x}_1, ..., \mathbf{x}_{N_u}) \quad \text{such that} \quad \mathcal{H}\Psi = E_{\min}\Psi, \tag{3.50}$$

$$\frac{\partial}{\partial x_{p,i}} E_{\min}(\mathbf{x}_1, ..., \mathbf{x}_{N_u}), \qquad \mathbf{x}_p = (x_{p,1}, x_{p,2}, x_{p,3}). \tag{3.51}$$

As an example of using the ATK simulation, we observe the isosurface of the electron density function in Fig. 4 for a composite system of germanium/silicon in which only the atoms inside the unit cell are displayed. The displayed figure corresponds to Ge_xSi_{1-x} where $x = 50\%$ in a germanium-silicon combination obtained from a diamond structure. After applying the approach described in the previous sections to several atomic configurations, we collect the major results on Table 3.3. It mainly summarizes the elapsed time for the preparation and the performance of the new method. In the tabulated outcomes, we consider first Body Centered Cubic of germanium. In addition, we use composites which are labeled Ge_xSi_{1-x} where $x \in [0, 1]$ controls the amounts of germanium and silicon. More precisely, we consider the composites which are $Ge_{0.75}Si_{0.25}$, $Ge_{0.50}Si_{0.50}$ and $Ge_{0.25}Si_{0.75}$. All those composites are structured by using Face Centered Cubic as space group symmetry. The appropriate parameter values are obtained from the American Mineralogist Crystal Structure Database [1].

The preparation step consists of a geometry optimization and the determination of the kernel approximation. The duration of the Gaussian kernel is dominated by the DFT computations related to point samples. The numbers of point samples are 70, 105, 200 and 250 for isotropic, 2D anisotropic, 3D anisotropic and Voigt tensor respectively. The stochastic computation as described in section 2.3 is very fast. In fact, the application of stochastic simulation is at most 2 percent of the whole preparation. All the computations were performed with the DFT basis unpolarized single ζ which is the least intensive basis available in the implementation. If other bases were used, the time for the DFT computation would last even much longer. In the case of DFT double ζ polarized, the scaling of the computation intensity might be doubled. In all dimensions, the preparation expense has long durations. Depending on the tensor transformation and the number of sample points, the preparation overhead can last a few minutes till several days. But the output of those preparations can be stored so that they need only be computed once for all. The ratio between the direct DFT-evaluation and the evaluation using kernel approximation is displayed in the last column of the Table. It exhibits that in average, the orders of acceleration are respectively $2.62e-06$, $9.16e-06$, $3.00e-05$ and $4.22e-05$ for the isotropic, $2D$ anisotropic, $3D$ anisotropic and $6D$ tensor cases. Since the acceleration advantage is very good, investing on the preparation process is worth calculating as the results can be stored and subsequently post-processed. In order to observe the improvement of the accuracy, we consider two silicon configurations in Figure 5. As a matter of fact, the first configuration consists of a silicon admitting a hexagonal lattice using

		PREPARATION		EVALUATIONS		
CONFIG.	DIM	Geom. opt.	Kernel setup	Direct DFT	Kernel	Ratio
BCC Ge	1D	4.791 mn	58.788 mn	89.116 mn	0.0238 sc	4.45e-06
	2D	8.632 mn	78.076 mn	105.952 mn	0.1257 sc	1.97e-05
	3D	13.353 mn	134.019 mn	182.194 mn	0.5181 sc	4.73e-05
	6D	58.238 mn	6.986 hr	8.346 hr	0.9142 sc	3.04e-05
FCC $Ge_{.75}Si_{.25}$	1D	57.627 mn	128.618 mn	172.634 mn	0.0272 sc	2.62e-06
	2D	67.437 mn	176.058 mn	229.228 mn	0.0776 sc	5.64e-06
	3D	139.958 mn	276.281 mn	363.696 mn	0.7345 sc	3.36e-05
	6D	7.493 hr	107.328 hr	143.634 hr	0.8964 sc	1.73e-06
FCC $Ge_{.50}Si_{.50}$	1D	33.119 mn	137.369 mn	205.144 mn	0.0225 sc	1.82e-06
	2D	79.887 mn	161.299 mn	212.237 mn	0.0710 sc	5.57e-06
	3D	104.440 mn	279.807 mn	377.709 mn	0.4704 sc	2.07e-05
	6D	7.896 hr	117.347 hr	133.213 hr	0.8978 sc	1.87e-06
FCC $Ge_{.25}Si_{.75}$	1D	43.675 mn	144.608 mn	201.094 mn	0.0193 sc	1.59e-06
	2D	71.893 mn	186.089 mn	233.715 mn	0.0807 sc	5.75e-06
	3D	159.021 mn	395.511 mn	498.650 mn	0.5624 sc	1.87e-05
	6D	7.962 hr	89.853 hr	110.410 hr	0.8994 sc	1.35e-04
AVERAGE	1D	34.803 mn	117.346 mn	166.997 mn	0.0232 sc	2.62e-06
	2D	56.962 mn	150.380 mn	195.283 mn	0.0887 sc	9.16e-06
	3D	104.192 mn	271.404 mn	355.562 mn	0.5713 sc	3.00e-05
	6D	6.080 hr	80.3785 hr	98.900 hr	0.9019 sc	4.22e-05

TABLE 3.3. Preparation overhead and acceleration gain.

space group $P6_3/mmc$. The second one admits a Face Centered Cubic lattice possessing the space group $Fd3m$. For both configurations, the bond lengths are obtained from the American Mineralogist Crystal Structure Database [1]. They are respectively represented by triangular and circular marks on the diagonal. That figure displays a comparison of the accuracy between the DFT computation and the kernel approximation. Marks which are closer to the diagonal identify good agreements between the two methods. We present only the case of anisotropic 3D transformation in this comparison because the other cases would exhibit similar results. The only difference between the two figures is that more

sample points are used in the second one. More precisely, 45 sampling points are used in the first simulation while 225 are used for the second one. One can observe that the values on the diagonals become more precise as more sampling points are used. Additional sampling points can be used if a better accuracy is desired with the costs of having longer preparation overhead.

As an outlook, we would like to mention some possible future enhancements. In Table 3.3, we consider only four configurations but our ultimate objective is to assemble a database of configuration to facilitate data retrieval and evaluation. The current results contain only germanium and silicon admitting a few space group structure. We will present subsequently some results about configurations admitting other crystal structures. In addition, some atomic elements and composites will be computed. We will consider also configurations containing defects. It is possible to use gradient and higher derivative information which will be deferred to a subsequent paper. Only non-adaptive cases are presented so far but it is possible to use the variance of the prediction as error estimators which enable adaptivity.

Acknowledgments

This work was partially supported by Eurostars Project E!6935 funded by German Federal Ministry of Education and Research.

References

[1] AMERICAN MINERALOGIST CRYSTAL STRUCTURE DATABASE, (http://rruff.geo.arizona.edu/AMS/)

[2] ATOMISTIK TOOLKIT VERSION 13.8.0, QuantumWise A/S (www.quantumwise.com).

[3] M. BRANDBYGE, J. MOZOS, P. ORDEJON, J. TAYLOR, K. STOKBRO, Phys. Rev. B **65**, 165401 (2002).

[4] J. HILL, L. SUBRAMANIAN, A. MAITI, *Molecular modeling techniques in material sciences*. Taylor & Francis Group (2005).

[5] W. KOHN, L. SHAM, *Self-consistent equations including exchange and correlation effects*, Phys. Rev. **140**, 4A, A1133-A11388 (1965).

[6] J. NOCEDAL, S. WRIGHT, *Numerical optimization*, Springer Series in Operation Research, New York (1999).

[7] L. GENOVESE, B. VIDEAU, M. OSPICI, T. DEUTSCH, S. GOEDECKER, J. MÉHAUT, *Daubechies wavelets for high performance electronic structure calculations: The BigDFT project Ondelettes de Daubechies pour le calculs de structure électronique à haute performance : Le code BigDFT*. Comptes Rendus Mécanique **339**, 149-164 (2011).

[8] S. JOHNSON, *The NLopt nonlinear-optimization package*, http://ab-initio.mit.edu/nlopt.

[9] J. PERDEW, Y. WANG, *Accurate and simple analytic representation of the electron-gas correlation energy*, Phys. Rev. B **45**, No. 23 (1992).

[10] J. PERDEW, A. ZUNGER, *Self-interaction correction to density-functional approximation for many-electron systems*, Phys. Rev. B **23**, 5048–5079 (1981).

[11] C. RASMUSSEN, C. WILLIAMS, *Gaussian processes for machine learning*, the MIT Press (2006).

[12] Y. SAAD, J. CHELIKOWSKY, S. SHONTZ, *Numerical methods for electronic structure calculations of materials*, SIAM Review **52**, No. 1, 3–54 (2010).

[13] R. SCHNEIDER,T. WEBER, *Wavelets for density matrix computation in electronic structure calculation*, Appl. Numer. Math. **56**, No. 10-11, 1383–1396 (2006).

[14] J. SOLER, E. ARTACHO, J. GALE, A. GARCIA, J. JUNQUERA, P. ORDEJON, D. SANCHEZ-PORTAL, *The SIESTA method for ab initio order-N materials simulation*, J. Phys. Condens. Matter **14**, 2745-2779 (2002).

[15] K. STOKBRO, D. PETERSEN, S. SMIDSTRUP, A. BLOM, M. IPSEN, K. KAASBJERG. *Semi-empirical model for nano-scale device simulations*, Phys. Rev. B **82**, 075420 (2010).

[16] P. SURYANARAYANA, V. GAVINI, T. BLESGEN, K. BHATTACHARYA, M. ORTIZ, *Non-periodic finite-element formulation of Kohn-Sham density functional theory*, Journal of the Mechanics and Physics of Solids **58**, 256-280 (2010).

[17] N. TODA, H. KIMIZUKA, S. OGATA, *DFT-based FEM analysis of nonlinear effects on indentation process in diamond crystal*, International Journal of Mechanical Sciences **52**, No. 2, 303–308 (2010).

[18] S. VOSKO, L. WILK, M. NUSAIR, *Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis*. Can. J. Phys. **58**, No. 8, 1200 (1980).

[19] E. WIMMER, H. KRAKAUER, M. WEINER, A. FREEMAN, *Full-potential self-consistent linearized-augmented-plane-wave method for calculating the electronic structure of molecules and surfaces: O_2 molecule*, Phys. Rev. B **24**, 864 (1981).

[20] L. LIN, C. YANG, J. MEZA, J. LU, L. YING, W. E, *SelInv-An algorithm for selected inversion of a sparse symmetric matrix*, ACM Trans. Math. Software **37**, No. 40 (2011).