

A self-consistent surface Green-function (SSGF) method for the calculation of isolated adsorbate atoms on a semi-infinite crystal

Jörg Bormet, Bernd Wenzien and Matthias Scheffler

Fritz-Haber-Institut der Max-Planck-Gesellschaft, Faradayweg 4–6, D-14195 Berlin (Dahlem), Germany

Received 16 September 1993

This paper describes a computer code for total-energy and electronic structure calculations of an isolated adsorbate atom on a metal surface using a self-consistent surface Green-function method. The approach is based on density-functional theory together with the local-density approximation of the exchange-correlation functional. The method allows to calculate the adsorption problem on an atomistic, semi-infinite metal surface. The code does not rely on particular symmetries of the surface or adsorbate, but it requires to have the Green function of a so-called “reference system” projected onto a set of Gaussian orbitals as an input. As a typical example, we present results for the adsorption of a single silicon atom at the fcc-hollow site on Al(111).

PROGRAM SUMMARY

Title of program: fhi93ssgf

Catalogue number: ACPV

Program obtainable from: CPC Program Library, Queen's University of Belfast, N. Ireland (see application form in this issue)

Licensing provisions: Persons requesting the program must sign the standard CPC non-profit use license (see license agreement printed in every issue).

Computer, operating system, and installation:

- IBM RISC System/6000; AIX 3.2; FHI der Max-Planck-Gesellschaft, Berlin.
- CRAY Y-MP/4; UNICOS 7.0; IPP der Max-Planck-Gesellschaft, Garching.
- AXP Alpha; OSF/1; FHI der Max-Planck-Gesellschaft, Berlin.

Programming language used: FORTRAN77, ANSI X3.9-1978 (exceptions from the standard are the use of IMPLICIT NONE and DOUBLE COMPLEX). The code is divided into five libraries for which makefiles on UNIX machines are available.

Floating point arithmetic: 64 bits

Memory required to execute with typical data: 50 Mbytes

Size of distributed program, including test data, etc.: approximately low files, about 20 Mbytes

Keywords: Green function, surface, adsorbate, layer KKR, DFT-LDA, total energy, Hellmann–Feynman forces

Nature of physical problem

The computer code allows to calculate the Green function of an adsorption problem with a single, isolated adsorbate atom (so-called “adsorbate system”) on a semi-infinite metal surface. The following physical quantities are available as output: change in electron density for the adsorbate system, change in density of states, total energy of the adsorbate system, and the Hellmann–Feynman forces on the adsorbate atom. The program uses density-functional theory within the local-density approximation for the exchange-correlation functional and *ab initio*, norm-conserving pseudopotentials.

Correspondence to: B. Wenzien, Friedrich-Schiller-Universität Jena, Institut für Festkörpertheorie und Theoretische Optik, Max-Wien-Platz 1, D-07743 Jena, Germany (present address).
E-mail:wenzien@ifdo.physik.uni-jena.de.

Method of solution

The Green function of the clean substrate (so-called “reference system”) has to be calculated in advance. This reference Green function is needed as input for this code. It is obtainable with the layer KKR method [1–3] and has to be projected onto a localized basis of Gaussian orbitals [4,5]. The code described below then solves the Dyson equation self-consistently for the effective potential of the adsorbate atom with the projected Green function of the reference system.

Restrictions on the complexity of the problem

At this time, only one single adsorbate atom can be handled by the code, although the input is made for a finite number of adsorbate atoms. For the evaluation of the exchange-correlation functional the electron-density change, $\Delta n^v(r)$, is evaluated on a mesh in real space. This mesh is restricted to be of cubic shape. The treatment of f-electron systems is not possible with the present code, although there are no limitations in principle.

Typical running time

One iteration on a CRAY Y-MP (single processor) takes 82 seconds, on an IBM RS/6000-350 it takes 493 seconds. About 40 iterations are necessary to converge a typical problem which has a linear dimension of 108 in the Gaussian basis and about 40^3 points in the real-space mesh for $\Delta n^v(r)$. There are three time-consuming parts:

- solution of the Dyson equation;
- projection of the effective potential onto Gaussians;
- transformation of the density matrix (in the Gaussian basis) to the real space mesh.

References

- [1] F. Máca and M. Scheffler, *Comput. Phys. Commun.* 38 (1985) 403; 47 (1987) 349.
- [2] F. Máca and M. Scheffler, *Comput. Phys. Commun.* 51 (1988) 381.
- [3] B. Wenzien, J. Bormet and M. Scheffler, *Green function for crystal surfaces I*, *Comput. Phys. Commun.*, submitted.
- [4] Ch. Droste, Ph. D. thesis, Fachbereich 4 (Physik) der Technischen Universität Berlin (1990).

LONG WRITE-UP**1. Introduction**

Our aim is to solve the adsorption problem of a single adatom on a semi-infinite crystal with an atomistic structure. The Kohn–Sham Hamiltonian [1,2] consists of the kinetic energy and the effective potential V^{eff} :

$$H = -\frac{1}{2}\nabla^2 + V^{\text{eff}} = -\frac{1}{2}\nabla^2 + V^{\text{ion}} + V^{\text{H}} + V^{\text{xc}}. \quad (1)$$

We use atomic units and measure the energy in hartrees (1 hartree = 27.211562 eV) and the length in bohrs (1 bohr = 0.052917706 nm). The effective potential is split into three parts. The first one is the external potential of the ionic cores,

$$\begin{aligned} V^{\text{ion}}(\mathbf{r}, \mathbf{r}') &= \sum_k (V_k^{\text{ion}, l_k^{\text{max}}}(|\mathbf{r} - \mathbf{R}_k|)\delta(\mathbf{r} - \mathbf{r}') + V_k^{\text{nl}}(\mathbf{r} - \mathbf{R}_k, \mathbf{r}' - \mathbf{R}_k)) \\ &= \sum_k \left(V_k^{\text{ion}, l_k^{\text{max}}}(|\mathbf{r} - \mathbf{R}_k|)\delta(\mathbf{r} - \mathbf{r}') \right. \\ &\quad + \sum_{l=0}^{l_k^{\text{max}}-1} \sum_{m_l=-l}^l \frac{V_k^{\text{ion}, l}(|\mathbf{r} - \mathbf{R}_k|) - V_k^{\text{ion}, l_k^{\text{max}}}(|\mathbf{r} - \mathbf{R}_k|)}{|\mathbf{r} - \mathbf{R}_k|^2} \\ &\quad \left. \times Y_{l, m_l}(\widehat{\mathbf{r} - \mathbf{R}_k}) Y_{l, m_l}^*(\widehat{\mathbf{r}' - \mathbf{R}_k}) \delta(|\mathbf{r} - \mathbf{R}_k| - |\mathbf{r}' - \mathbf{R}_k|) \right), \end{aligned}$$

where k counts the atoms and which are represented by ab initio, norm-conserving pseudopotentials and are thus non-local operators in real space. The second term in eq. (1) is the Hartree potential $V^H(\mathbf{r}) = \int [n^v(\mathbf{r}')/|\mathbf{r} - \mathbf{r}'|] d^3r'$, and the third one is the exchange-correlation potential $V^{xc}(\mathbf{r})$ which is treated within the LDA [3,4].

The ionic pseudopotentials are available from the tables of Bachelet, Hamann, and Schlüter [5] in the following form

$$V_k^{\text{ion},l}(\mathbf{r}) = V_k^{\text{core}}(\mathbf{r}) + \Delta V_k^{\text{ion},l}(\mathbf{r}) = -\frac{Z_k^v}{r} \left(\sum_{i=1}^2 c_k^i \text{erf}(\sqrt{\alpha_k^i} r) \right) + \sum_{i=1}^3 (a_k^{i,l} + r^2 b_k^{i,l}) e^{-\beta_k^{i,l} r^2}. \quad (3)$$

Elements with $Z < 55$ have a non-local expansion with a maximum angular momentum of $l_k^{\text{max}} = 2$.

2. The self-consistent surface Green-function method

In this section we describe the basic ideas of the method. Applications of this method to the adsorption of Na, Si, and Cl on Al and Cu surfaces have already been published [6–8]. It is convenient to split the Hamiltonian of the total system (as well as the electron density $n^v(\mathbf{r})$) into two parts,

$$H = H^0 + \Delta V^{\text{eff}}. \quad (4)$$

The “reference system” $H^0 = -\frac{1}{2}\nabla^2 + V^0$ describes the two-dimensionally periodic, semi-infinite substrate for a given potential V^0 which is assumed to be self-consistent in the bulk region and has a step barrier at the surface. The height of this barrier should be determined by the bulk Fermi level and the clean surface work function. Whereas the Fermi level is most efficiently obtained from a bulk calculation, the work function should be calculated using a slab approach. The electron density which follows from H^0 is denoted as $n^{v,0}(\mathbf{r})$. The potential V^0 is correct 1–2 interatomic distances away from the surface, but directly at the surface it is only an approximation. Therefore, the self-consistent change, ΔV^{eff} , accounts not only for the adsorbed atom but also for the contributions which arise to make the surface locally self-consistent.

Splitting the total Hamiltonian as done in eq. (4) leads to an *operator* Green-function method which has to treat the operator of the kinetic energy only in the reference system. Besides the localization of ΔV^{eff} the only assumption of the method is that the change of the valence electron density,

$$\Delta n^v(\mathbf{r}) = n^v(\mathbf{r}) - n^{v,0}(\mathbf{r}), \quad (5)$$

of the adsorbate system is localized in real space. In particular for metals, the screening properties guarantee that this is fulfilled usually. Although the electron-density *change* is localized in real space we emphasize that the adsorbate wave functions are typically extended, which is taken into account without any constraint.

For the change of the effective potential of the adsorbate system we write

$$\Delta V^{\text{eff}} = \Delta V^{\text{ion}} + \Delta V^H + \Delta V^{xc} + \Delta V^{\text{eff,sc,o}}, \quad (6)$$

where the first term on the right is the change due to the ionic potential of the adsorbate, the second term is due to the Hartree potential and the third term describes the change of the exchange-correlation potential, $\Delta V^{xc}(\mathbf{r}) = V^{xc}[\Delta n^v(\mathbf{r}) + n^{v,0}(\mathbf{r})] - V^{xc}[n^{v,0}(\mathbf{r})]$. The last term in eq. (6) accounts for the local contributions towards self-consistency of the clean surface, $\Delta V^{\text{eff,sc,0}} = V^{\text{eff,0}} - V^0$.

The Green function of the reference system, $G^0(\mathcal{Z}) = (\mathcal{Z} - H^0)^{-1}$, is calculated for all needed complex energies $\mathcal{Z} = \epsilon + i\eta$ as described in refs. [9,10]. Once G^0 is known, the Green operator of the total system, $G(\mathcal{Z})$, is given by the Dyson equation,

$$G(\mathcal{Z}) = G^0(\mathcal{Z}) + G^0(\mathcal{Z})\Delta V^{\text{eff}}G(\mathcal{Z}). \quad (7)$$

Taking advantage of the localization of $\Delta V^{\text{eff}}(\mathcal{Z})$ the *change* of the Green function can be calculated efficiently from

$$\Delta G(\mathcal{Z}) = G(\mathcal{Z}) - G^0(\mathcal{Z}) = G^0(\mathcal{Z})\Delta V^{\text{eff}}(1 - G^0(\mathcal{Z})\Delta V^{\text{eff}})^{-1}G^0(\mathcal{Z}). \quad (8)$$

With the projection operator $P = \sum_{i,j} |\chi_i\rangle\langle\chi_j|$, defined by local basis functions which span the real-space region of $\Delta n^v(\mathbf{r})$ and $\Delta V^{\text{eff}}(\mathbf{r})$, we have the identity

$$Pn^vP = n^v. \quad (9)$$

The electron-density change is then given by

$$\Delta n^v(\mathbf{r}) = -\frac{2}{\pi} \text{Im} \left(\oint_{\Gamma^\epsilon} \langle \mathbf{r} | P \Delta G(\mathcal{Z}) P | \mathbf{r} \rangle d\mathcal{Z} \right). \quad (10)$$

The factor 2 takes the spin degeneracy into account. In order to evaluate eq. (10) we use an energy contour Γ^ϵ in the complex energy plane which includes all occupied valence states. Thus, we start at an energy ϵ^{bot} clearly below the substrate valance band, go into the complex energy plane and come back to the real axis at the substrate Fermi energy, ϵ_F . The quantity $P\Delta G(\mathcal{Z})P$ is evaluated from

$$P\Delta G(\mathcal{Z})P = PG^0(\mathcal{Z})P\Delta V^{\text{eff}}P(1 - PG^0(\mathcal{Z})P\Delta V^{\text{eff}}P)^{-1}PG^0(\mathcal{Z})P, \quad (11)$$

which requires only localized quantities. The needed projection $PG^0(\mathcal{Z})P$ is taken from refs. [10,11].

In a first step one can take the atomic electron density of the adsorbate atoms as a start guess, calculating the effective potential thereof, solving eq. (11) and again extracting an electron density with eq. (10) for the next iteration cycle until this process is converged.

3. Total energies and Hellmann–Feynman forces

Once the valence electron density of the adsorbate system, $\Delta n^v(\mathbf{r})$, is calculated we can evaluate the total energy of the adsorbate system. The total-energy functional for the valence electron density $n^v(\mathbf{r})$, which describes *all* the valence electrons of the substrate and the adsorbate, can be written as

$$\begin{aligned} E^{\text{tot}}[n^v] = & \sum_{i=1}^{\text{occ}} \epsilon_{v,i}^{\text{out}} + \frac{1}{2} \iint \frac{n^{\text{v,out}}(\mathbf{r})n^{\text{v,out}}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r d^3r' - \iint \frac{n^{\text{v,out}}(\mathbf{r})n^{\text{v,in}}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r d^3r' \\ & + \int (n^{\text{v,out}}(\mathbf{r}) + n^{\text{c,at}}(\mathbf{r}))\epsilon^{\text{xc}}[n^{\text{v,out}}(\mathbf{r}) + n^{\text{c,at}}(\mathbf{r})] d^3r \\ & - \int n^{\text{v,out}}(\mathbf{r})V^{\text{xc}}[n^{\text{v,in}}(\mathbf{r}) + n^{\text{c,at}}(\mathbf{r})] d^3r + \frac{1}{2} \sum'_{k,l} \frac{\mathbf{Z}_k^v \mathbf{Z}_l^v}{|\mathbf{R}_k - \mathbf{R}_l|} + \epsilon_F dN. \end{aligned} \quad (12)$$

The indices “out” and “in” indicate if the corresponding quantity is input respectively output of one specific iteration of the Kohn–Sham equation (in our case the Dyson equation) and take care of the correct handling of the variational principle which underlies the total energy of eq. (12). ϵ^{xc} stands for

the exchange-correlation energy per electron of the LDA. V^{xc} is the corresponding exchange-correlation potential. Although we use the frozen-core approximation to find ourselves in the pleasant situation to be able to use the scheme of pseudopotentials we do not linearize the exchange-correlation functional between the density of core and valence electrons. This immediately leads to the electron density of the cores, $n^{\text{c,at}}(\mathbf{r})$, to be present in eq. (12). The contribution $\frac{1}{2}\sum_{k,l} Z_k^v Z_l^v / |\mathbf{R}_k - \mathbf{R}_l|$ is the electrostatic interaction of the ionic cores. The last term, $\epsilon_F dN$, accounts for the fact that our Green-function method uses a constant chemical potential ϵ_F – the Fermi level – for the integration in eq. (10) which implies that there is no constraint for charge neutrality in the system. The charge discrepancy is characterized by dN (in all practical cases of adsorption we find dN to be smaller than one-hundredth of an electron). Therefore, we actually minimize the Gibbs free energy if we use E^{tot} from eq. (12).

Now it is an important fact to recognize that the total energy of the adsorbate system can be evaluated with the knowledge of $\Delta n^v(\mathbf{r})$ in a spatial region Ω^{loc} in which $\Delta n^v(\mathbf{r})$ is different from zero. This is true because all the kernels of the integrals in eq. (12) can be factorized into products of two quantities of which one is strongly localized in that spatial region Ω^{loc} . Nevertheless, it is necessary to know the valence electron density, $n^{v,0}(\mathbf{r})$, of the reference system in Ω^{loc} .

If we take into account that all integrals must be exactly evaluated – in the same manner as they appear in the term of the single-particle energy (with respect to the spatial localization) – the total energy of the adsorbate system reads

$$\begin{aligned}
\Delta E^{\text{tot}} = & \frac{2}{\pi} [\epsilon_F \delta^{\text{out}}(\epsilon_F) - \epsilon^{\text{bot}} \delta^{\text{out}}(\epsilon^{\text{bot}})] + \frac{2}{\pi} \oint_{\Gamma^e} \delta^{\text{in}}(\mathcal{Z}) d\mathcal{Z} \\
& + \frac{1}{2} \int_{\Omega^{\text{loc}}} \Delta n^{v,\text{out}}(\mathbf{r}) \left(\int_{\Omega^{\text{loc}}} \frac{\Delta n^{v,\text{out}}(\mathbf{r}') - n^{\text{tst},\text{out}}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3 r' + V^{\text{tst},\text{out}}(\mathbf{r}) \right) d^3 r \\
& - \int_{\Omega^{\text{loc}}} \Delta n^{v,\text{out}}(\mathbf{r}) \left(\int_{\Omega^{\text{loc}}} \frac{\Delta n^{v,\text{in}}(\mathbf{r}') - n^{\text{tst},\text{in}}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3 r' + V^{\text{tst},\text{in}}(\mathbf{r}) \right) d^3 r \\
& + \int_{\Omega^{\text{loc}}} \left[\sum_{a=1}^{N^{\text{ad}}} n_a^{\text{ion}}(|\mathbf{r} - \mathbf{R}_a|) \right] \left(\int_{\Omega^{\text{slab}}} \frac{\left(\sum_{l=N^{\text{ad}}+1}^{\infty} n_l^{\text{ion}}(|\mathbf{r}' - \mathbf{R}_l|) \right) - n^{v,0}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3 r' \right) d^3 r \\
& + \int_{\Omega^{\text{loc}}} \left[\text{Im} \left(\sum_{i,j} (S^{-1} n^{v,0} S^{-1})_{ij} \chi_i(\mathbf{r}) \chi_j^*(\mathbf{r}) \right) \right] \\
& \times \left(\int_{\Omega^{\text{loc}}} \frac{\sum_{a=1}^{N^{\text{ad}}} n_a^{\text{ion}}(|\mathbf{r}' - \mathbf{R}_a|)}{|\mathbf{r} - \mathbf{r}'|} d^3 r' + \int_{\Omega^{\text{loc}}} \frac{n^{\text{tst},\text{in}}(\mathbf{r}') - \Delta n^{v,\text{in}}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3 r' - V^{\text{tst},\text{in}}(\mathbf{r}) d^3 r' \right) d^3 r \\
& + \int_{\Omega^{\text{loc}}} (\Delta n^{v,\text{out}}(\mathbf{r}) + n^{v,0}(\mathbf{r}) + n^{\text{c,at}}(\mathbf{r})) \\
& \times (\epsilon^{\text{xc}} [\Delta n^{v,\text{out}}(\mathbf{r}) + n^{v,0}(\mathbf{r}) + n^{\text{c,at}}(\mathbf{r})] - \epsilon^{\text{xc}} [n^{v,0}(\mathbf{r}) + n^{\text{c,at},0}(\mathbf{r})]) d^3 r \\
& + \int_{\Omega^{\text{loc}}} (\Delta n^{v,\text{out}}(\mathbf{r}) + \Delta n^{\text{c,at}}(\mathbf{r})) \epsilon^{\text{xc}} [n^{v,0}(\mathbf{r}) + n^{\text{c,at},0}(\mathbf{r})] d^3 r
\end{aligned}$$

$$\begin{aligned}
& - \int_{\Omega^{\text{loc}}} (\Delta n^{\text{v,out}}(\mathbf{r}) + n^{\text{v},0}(\mathbf{r})) \times (V^{\text{xc}}[\Delta n^{\text{v,in}}(\mathbf{r}) + n^{\text{v},0}(\mathbf{r}) + n^{\text{c,at}}(\mathbf{r})] \\
& - V^{\text{xc}}[n^{\text{v},0}(\mathbf{r}) + n^{\text{c,at},0}(\mathbf{r})]) d^3r - \int_{\Omega^{\text{loc}}} \Delta n^{\text{v,out}}(\mathbf{r}) V^{\text{xc}}[n^{\text{v},0}(\mathbf{r}) + n^{\text{c,at},0}(\mathbf{r})] d^3r \\
& + \sum_{a=1}^{N^{\text{ad}}} \sum_{b=a+1}^{N^{\text{ad}}} \frac{Z_a^{\text{v}} Z_b^{\text{v}}}{|\mathbf{R}_a - \mathbf{R}_b|} + \epsilon_{\text{F}} \left(\sum_{a=1}^{N^{\text{ad}}} Z_a^{\text{v}} - Q^{\text{n}} \right). \tag{13}
\end{aligned}$$

The spatial region Ω^{slab} has the planar translational symmetry of the reference system and a thickness of several vacuum and crystal layers – at least as many as they are included in Ω^{loc} . Equation (13) exhibits only a few additional new symbols which are not defined so far. One of them is the generalized phase shift defined by

$$\delta(\mathcal{Z}) = -\text{Im} \ln \det(I - G^0(\mathcal{Z}) S^{-1} \Delta V^{\text{eff}} S^{-1}). \tag{14}$$

In practice, we use a basis set of Gaussian orbitals,

$$\chi_i(\mathbf{r}) = |\mathbf{r} - \mathbf{O}_i|^{l_i} e^{\kappa_i |\mathbf{r} - \mathbf{O}_i|} Y_{l_i, m_{l_i}}(\widehat{\mathbf{r} - \mathbf{O}_i}), \tag{15}$$

to define the projection operator of eq. (9). \mathbf{O}_i is called the orbital site, κ_i is the decay constant and l_i is the angular momentum of the Gaussian orbital i . Because the Gaussian orbitals are not orthonormal we have to insert the inverse S^{-1} of the overlap matrix (elements are defined by $S_{ij} = \int_{\Omega^{\text{loc}}} \chi_i^*(\mathbf{r}) \chi_j(\mathbf{r}) d^3r$) wherever the projection operator P occurs in the formulas of the previous section. I is the matrix of unity. N^{ad} counts the (finite) number of adsorbate atoms whereas n_i^{ion} gives the charge distribution of the ionic cores of the pseudopotentials.

The quantities $n^{\text{tst}}(\mathbf{r})$ are test charge densities with the property that the difference $\Delta n^{\text{v}}(\mathbf{r}) - n^{\text{tst}}(\mathbf{r})$ has no monopole and dipole moment. This is useful to carry out the corresponding integrals with the technique of the fast Fourier transform. Furthermore, the test charges should provide a mathematical representation which allows the evaluation of the Poisson equation, $\nabla^2 V^{\text{tst}}(\mathbf{r}) = -4\pi n^{\text{tst}}(\mathbf{r})$, analytically.

In one of the integrals, the electron density of the reference system,

$$\text{Im} \left(\sum_{i,j} (S^{-1} n^{\text{v},0} S^{-1})_{ij} \chi_i(\mathbf{r}) \chi_j^*(\mathbf{r}) \right), \tag{16}$$

with its density matrix $n^{\text{v},0}$ is calculated in the localized representation to make sure that the term is treated exactly as it contributes to the single-particle energy.

If one is interested in the Hellmann–Feynman forces one has to compute the total derivative $\mathbf{F}_m = -dE^{\text{tot}}/d\mathbf{R}_m$ of the energy functional with respect to the position of the m th nuclei. Similar to the energy functional the integrals have kernels with two factors of which at least one is localized. With the abbreviation $\bar{\mathbf{r}}_m = |\mathbf{r} - \mathbf{R}_m|$ and the property $\partial \bar{\mathbf{r}}_m / \partial \mathbf{r}_m = -(\mathbf{r} - \mathbf{R}_m) / \bar{\mathbf{r}}_m^3$ the force for the m th nuclei of the adsorbate atoms can be written as

$$\begin{aligned}
\mathbf{F}_m = & \int_{\Omega^{\text{loc}}} \frac{\partial V_m^{\text{core}}(\bar{\mathbf{r}}_m)}{\partial \bar{\mathbf{r}}_m} (\mathbf{r} - \mathbf{R}_m) \left[\Delta n^{\text{v,out}}(\mathbf{r}) - \sum_{a=1}^{N^{\text{ad}}} n_a^{\text{ion}}(|\mathbf{r} - \mathbf{R}_a|) \right] d^3r \\
& + \int_{\Omega^{\text{loc}}} \frac{\partial n_m^{\text{ion}}(\bar{\mathbf{r}}_m)}{\partial \bar{\mathbf{r}}_m} (\mathbf{r} - \mathbf{R}_m) \left(\int_{\Omega^{\text{slab}}} \frac{\left(\sum_{l=N^{\text{ad}}+1}^{\infty} n_l^{\text{ion}}(|\mathbf{r}' - \mathbf{R}_l|) \right) - n^{\text{v},0}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r' \right) d^3r
\end{aligned}$$

$$\begin{aligned}
& + \int_{\Omega^{\text{loc}}} \frac{\partial [\Delta V_m^{\text{ion}, L_m}(\bar{\mathbf{r}}_m)]}{\partial \bar{\mathbf{r}}_m} (\overline{\mathbf{r} - \mathbf{R}_m}) (\Delta n^{\text{v}, \text{out}}(\mathbf{r}) + n^{\text{v}, 0}(\mathbf{r})) d^3 r \\
& - \text{Im} \left[\text{Tr} \left(S^{-1} (\Delta n^{\text{v}, 0} + n^{\text{v}, 0}) S^{-1} \frac{\partial V^{\text{nl}}}{\partial \mathbf{R}_m} \right) \right] \\
& + \int_{\Omega^{\text{loc}}} V^{\text{xc}} [\Delta n^{\text{v}, \text{out}}(\mathbf{r}) + n^{\text{v}, 0}(\mathbf{r}) + n^{\text{c}, \text{at}}(\mathbf{r})] \frac{\partial n_m^{\text{c}, \text{at}}(\bar{\mathbf{r}}_m)}{\partial \bar{\mathbf{r}}_m} (\overline{\mathbf{r} - \mathbf{R}_m}) d^3 r.
\end{aligned} \tag{17}$$

The partial derivative of the matrix containing the non-local contributions of the pseudopotentials V^{nl} with respect to \mathbf{R}_m (see eq. (2)) is done with the centered differential quotient

$$\frac{\partial V^{\text{nl}}}{\partial \mathbf{R}_{m, \sigma}} = \lim_{h_\sigma \rightarrow 0} \frac{V^{\text{nl}}(\mathbf{R}_{m, \sigma} + h_\sigma) - V^{\text{nl}}(\mathbf{R}_{m, \sigma} - h_\sigma)}{2h_\sigma}, \quad \sigma = x, y, z. \tag{18}$$

4. Numerical details

Quantities with a local dependency on the vector \mathbf{r} (for example the change in the electron density $\Delta n^{\text{v}}(\mathbf{r})$) have a real-space representation on a three-dimensional regular cubic mesh with M mesh points and the discretization interval h . M is typically in the order of 10^5 and h is in the range from 0.2, ..., 0.6

Calculate projection ΔV^{ion} .	
Choose "arbitrary" $\Delta n^{\text{v}, i=0}(\mathbf{r})$ on real-space mesh.	
Calculate $\Delta V^i(\mathbf{r}) := \Delta V^{\text{H}}[\mathbf{r}, \Delta n^{\text{v}, i-1}(\mathbf{r})] + \Delta V^{\text{xc}}[n^{\text{v}, i-1}(\mathbf{r})]$.	
Projection of $\Delta V^i(\mathbf{r}) \rightarrow \Delta V^i$.	
$\Delta V^{\text{eff}, i} := \Delta V^i + \Delta V^{\text{ion}}$.	
Mixing: $\Delta V^{\text{eff}, i} := \alpha \Delta V^{\text{eff}, i-1} + (1 - \alpha) \Delta V^{\text{eff}, i}$.	
Solve: $\Delta G(\mathcal{Z}) =$ $G^\circ(\mathcal{Z}) S^{-1} \Delta V^{\text{eff}} S^{-1} (I - G^\circ(\mathcal{Z}) S^{-1} \Delta V^{\text{eff}} S^{-1})^{-1} G^\circ(\mathcal{Z})$.	
Calculate in real space $\Delta n^{\text{v}, i}(\mathbf{r})$.	
$\Delta V^{\text{eff}, i}$ converged?	
Postprocessing.	

Fig. 1. Iteration of the Dyson equation.

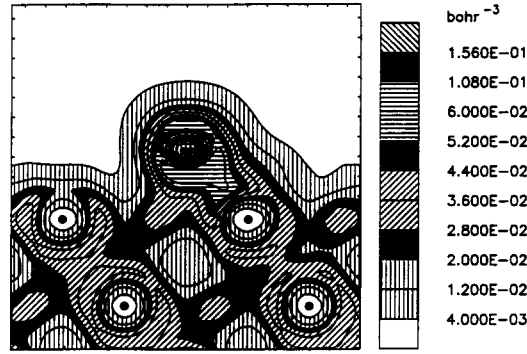


Fig. 2. Valence charge density for a single isolated Si atom adsorbed on the fcc-site of Al(111).

bohrs. Any integral over such functions is evaluated by summing up its weighted values at the mesh point, for example the charge represented by $\Delta n^v(\mathbf{r})$ is calculated from

$$q = \int_{\Omega^{\text{loc}}} \Delta n^v(\mathbf{r}) d^3r = \sum_{m=1}^M h^3 \delta n^v(\mathbf{r}_m). \quad (19)$$

Non-local quantities (dependent on \mathbf{r} and \mathbf{r}') are calculated analytically with formulas given in ref. [19].

The most important part of the program is the iteration loop. Figure 1 shows a flow diagram which can be directly mapped onto a corresponding loop in the main program. At first one has to do two things in advance: To calculate the projection of the fixed part of the effective potential – this is the projection of the pseudopotential – and to choose a rather arbitrary start electron density. The closer this density is to the converged result the less iterations are to be done. In practice, we provide a start guess of a Gaussian-like distribution with a charge of approximately the charge of the ionic core of the pseudopotential. Once the electron density $\Delta n^v(\mathbf{r})$ is available the Hartree potential is calculated with the method of the fast Fourier transform and the exchange-correlation potential is directly evaluated from the parametrization of Perdew and Zunger [4]. The next step is to project the two contributions onto Gaussian orbitals and to add the already projected pseudopotential to compile up the effective potential. If there is an effective potential from a previous iteration available it is mixed such that the new potential contributes with a percentage of about two to twenty per cent. This means that the mixing parameter α is

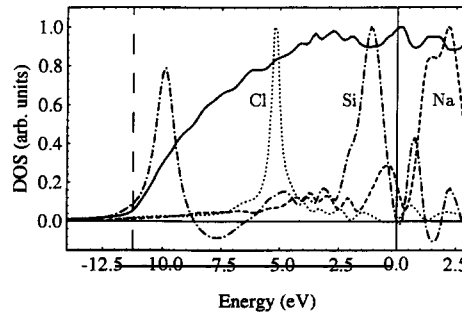


Fig. 3. Change in the density of states ($\text{Im } \mathcal{Z} = 0.2$ eV) induced by the adsorption of a single isolated Na (dashed line), Si (dashed-dotted line), and Cl (dotted line) atom on a Al(111) surface with respect to the density of states for the clean surface which is given by the solid line as a reference. The energy is referred to the Fermi level.

in the range of $0.80, \dots, 0.98$. The next step is to solve the Dyson equation which requires the inversion of a general complex matrix. After that the change in the electron density, $\Delta n^v(\mathbf{r})$, is calculated from eq. (10). If the effective potential is converged (the relative change of the norm of the matrices of the effective potential from two adjacent iterations is smaller than some tolerance $\tau \approx 10^{-3}$) the iterations are finished, otherwise the procedure is repeated with the new density as input.

5. Adsorption of a Si adatom on Al(111)

In order to explain a typical run of the program we have build up a test suite which treats the adsorption of a single isolated silicon adatom on the fcc-site of Al(111) [6,7]. The results, i.e., the valence charge density of the adsorbate system and the change in the density of states induced by the adsorption of the Si atom with respect to the density of states for the clean surface (compared with the corresponding quantities for Na and Cl adsorption as well), are shown in figs. 2 and 3. One requirement is to have a projected reference Green function. This file may be calculated in the manner of the example given in ref. [10]. In addition, it is provided as an ASCII file together with a short utility program which builds up the binary input file which is needed in the computation. All the other input files are part of the standard test suite stored at CPC.

The program is called `iter.x` and may be build by makefiles which are available for at least three different UNIX-like operating systems (UNICOS, AIX, OSF/1). To run the test suite the executable must be build first. The code is highly portable and might even run on a wide variety of different machines (including VAX under VMS). We decided to discuss directly an UNIX example script which runs the program. If one really needs to port it onto other architectures the most important information in the UNIX script are the numbers of input units. For example, a file named `fort.8` is connected to the FORTRAN unit 8 while the program is executing. It is useful to introduce explicit names for all files which are physically important. Therefore, we used the UNIX link command (`ln -s filename1 fort.1`) where `filename1` might be any valid name. Our naming convention is as follows: The abbreviation `si` means silicon, a file has the identifier `basis` in its name if the contained information depends on the Gaussian orbital basis. `v3c4` identifies that the projection of the reference Green function was done in a complex layer consisting of a vacuum region which is equivalent in thickness to that of three crystal sublayers and of four crystal sublayers. The example script is assumed to be in any subdirectory together with the executable `iter.x` and some other files which are described below.

6. Description of the input

The example can be executed just by running the provided UNIX shell script named `script.csh` and printed in the following:

```
#!/bin/csh-f
#=====
#   creation date: 1992-09-20 ----- author: Joerg Bormet
#       library: <none>-----processor: UNIX, csh
#----- abstract -----
#
#   Shell script for iteration of dyson equation.
#   Program: iter.x
#----- prepare flags -----
```

```

#
cat >iter.cn <<*EOI
=====
*   creation date: 1992-09-24 ----- author: Joerg Bormet
*       library: <none> ----- processor: DATA
*----- abstract -----
*
*   Flags for operations of program iter.x.
*
*----- used by iter -----
*
1:  5 * ITERMx
9:  0 * LHYDRO
10: 1 * ion-ion contribution
11: 1 * LAUTO
13: 0 * LDVCOE
14: 0 * LVSTRT
15: 0 * LRDRES
17: 0 * LPLOT
18: 1 * LWRITE
19: 0 * LRHOF: calculate (0), read (1)
20: 1 * read (0) or calculate (1) overlap matrix
21: 1 * local potential, vl (1), 1/r (2), erf/r (3)
22: 0 * forced compare of origin of input solution
23: 0 * calculation of flag 20 is done analytically
24: 97 * startmixing (percentage of input potential)
25: 90 * lower bound for mixing (percentage of inp. potential)
26: 99 * upper bound for mixing (percentage of inp. potential)
28: 0 * add delta G to reference
31: 5 * optimization interval
32: 0 * optimize decay
*
*-----
*EOI
#
#-----prepare filesystem -----
#
rm -f fort.3 fort.4 fort.7 fort.9 fort.10 fort.11 fort.21\
    fort.22 fort.23 fort.24 fort.40 fort.41 fort.42 fort.91\
    fort.96 fort.98 fort.99
#
ln -s v3c4basis,gr fort.4
# . . reference Green function on contour
#
ln -s ACT fort.8
# . . action control file
#

```

```

ln -s basis.si fort.11
# . . inverted overlap matrix
#
ln -s basis.cr fort.24
# . . electron density of reference system on real space mesh
#
ln -s basis.cd fort.96
# . . electron density of reference system as density matrix
#
ln -s si.ii fort.40
# . . energy and forces from ion-ion contribution
#
ln -s sibasis.is fort.9
# . . if old input solution is available, use it
#
ln -s sibasis.os fort.10
# . . output solution contains: density matrix, perturbation
#   potential and electron density
#
ln -s iter.cn fort.99
# . . flags used by iter.x
#
#-----iterations -----
#
iter.x < sibasis.d
# . . start program and read data from standard input
#
rm -f fort.3 fort.4 fort.7 fort.8 fort.9 fort.11 fort.10 fort.21\
    fort.22 fort.23 fort.24 fort.40 fort.41 fort.42 fort.91\
    fort.96 fort.98 fort.99
#
#
#-----

```

The execution of this shell script requires the existence of four files which must be provided in advance:

v3c4basis.gr (unit 4): Projected Green function of the reference system. This file may be generated in the manner of the example input of the program published in ref. [10].

sibasis.d (unit 5, standard input): Input data of the adsorbate system for the self-consistent iteration. This file will be discussed extensively below in this paper.

ACT (unit 8): This file contains parameters to influence the behaviour while **iter.x** is executing. One can change the numbers of iterations and the mixing parameters for the potential.

si.ii (unit 40): Contains the $\Delta n^v(r)$ -independent contributions for the total energy and the forces. This file is part of the distribution of the example discussed in this paper.

While the script is executing a couple of files are created (either by the script itself or by the execution of **iter.x**):

`iter.cn` (unit 99): Control file for the determination of program flow. It provides flags numbered with 1 to 50 which are translated to `INTEGERs` or `LOGICALs` within the code. The most important flags are:

- 1 Number of iterations to be done. In this example: five.
- 10 Is unit 40 expected as input? (Yes: 1, No: 0). In this example: Yes.
- 15 If there is a previous run, should the solution be read for further processing? (Yes: 1, No: 0). In this example: No.
- 19 Is the electron density and the density matrix of the reference system to be calculated (0) or to be read (1)? In this example: To be calculated.
- 20 Is the inverted overlap matrix, S^{-1} , to be calculated (1) or to be read (0)? In this example: To be calculated.
- 21 Should the program use the potential in the parametrization of eq. (2) as external potential? In this example: Yes.
- 24 Potential mixing in hundredth of the old potential. In this example: 97%.

After the end of the execution one will find some new files:

`basis.si` (unit 11): Inverted overlap matrix.

`basis.cr` (unit 24): Electron density, $n^{v,0}(\mathbf{r})$, calculated from the projected reference Green function.

`basis.cd` (unit 96): Density matrix corresponding to `basis.cr`.

`sibasis.os` (unit 10): Contains the density matrix of the adsorbate system, the matrix of the effective potential, $\Delta V^{\text{eff},\text{in}}$, and the electron density itself, $\Delta n^v(\mathbf{r})$.

Standard output: Messages about the program flow and the information about total energy (variable `ETOT` on the standard output) and force. Any message has the module name which created it in front.

If one likes to do further iterations with the program (which might be necessary to converge the solution) set flags 15 and 19 to one, flag 20 to zero and copy `sibasis.os` onto the file `sibasis.is` and then restart the shell script. The new run again does five iterations but uses the output of the first run (files `basis.si`, `basis.cr`, and `basis.cd` which are fixed and `sibasis.os` which was copied to `sibasis.is`).

All the input data related to physical quantities are collected in the file `sibasis.d` (standard input). The data records in this file are structured by groups where every group is enclosed in a `#Begin -` and `#End -` environment. In this sense, the whole data in the file itself represent such a group – the group called `#Begin/End-Of-Iter-Data-Set`. The groups are discussed in detail below. The real-space coordinates are given in (x, y, z) -tripels in units of the lattice constant of the underlying substrate. The $-z$ -direction is assumed to coincide with the outward surface normal. Names written in capital letters of a teletype are names of variables of the FORTRAN program. The file `sibasis.d` for the example given in this paper is printed at the end of the description of the groups.

#Begin/End-Of-Iter-Data-Set: At the level of this group the version of the data set (version is 2), the symmetry of the adsorbate system (this example: fcc(111)) and all the other groups are defined. For selecting the symmetry see also the `#Begin/End-Of-Box`-group below.

#Begin/End-Of-Gaussians: The Gaussian orbital basis becomes defined. The first of the three integer numbers, `NDIM` = 108, gives the linear dimension of the Gaussian basis. `NDEF` = 12 counts the lines in the `#Begin/End-Of-Sites`-group. `NBUFF` = 9 is divisor of `NDIM` = 108 and gives a size for a memory buffer used in the calculation of the real-space representation of the electron density and the projection of the effective potential. Small values of `NBUFF` stand for long computation time and less memory usage. Large values provide shorter computation time and large memory consumption. A very large memory usage may slow down the computation. Therefore set `NBUFF` to about ten to twenty per

cent of `NDIM`. In any line of the `#Begin/End-Of-Sites`-group one finds (from the left to the right) the three (x, y, z)-components of the orbital sites O_i , the decay constant κ_i , and an integer number for the largest value of the angular momentum which should be used together with each site and decay. A number of $l_k^{\max} = 2$ means to include s, p and d contributions.

`#Begin/End-Of-Box`: `NOCT` gives the discretization in one octant of the real space mesh which is always of cubic shape. The total number of mesh points, M , can be calculated from $M = (2 * \text{NOCT} + 1) * 3$. `RMRBOX` gives the basis vectors for the cubic mesh. This variable is included for further use and the values given in this example are the only possible values. `SPN = 21.693...` denotes the length of one edge of the cubic mesh. The origin of the cubic box is identical with the origin of the Gaussians. Because the cubic box may lower the symmetry of the substrate (as in the case of fcc(111)) not all of the possible symmetry transformations of the fcc(111) surface are used to speed up the calculation. Only the y - z mirror plane is included in this case. For fcc(001) all symmetry operations are used. In both cases, the input set of the Gaussian orbitals *and* the substrate must be oriented such that their symmetry axes are identical with those of the cubic mesh.

`#Begin/End-Of-Substrate`: The pointer "13" indicates that this example deals with an aluminum substrate. The pointer is an abbreviation for the whole group `#Begin/End-Of-Element` which is discussed below. The second number (2) tells the program to treat the d -potential of the pseudopotential as local one ($l_k^{\max} = 2$, compare with eq. (2)). The next quantities are the lattice constant given in bohrs, the Fermi-level, the muffin-tin level relative to the $(1/r)$ -asymptote of the local part of the potential, the work function of the substrate, and the z -position of the geometrical surface (jellium edge, if the substrate is jellium) relative to the origin of the Gaussian orbitals. Energies are given in electron volts (eV). `NGFBUF` determines the number of energy points which can be loaded at once into the central memory. The variation of this parameter has only a minor effect on the computation time (at maximum 10% were measured).

`#Begin/End-Of-Perturbations`: `NADSO` gives the number of adsorbate atoms, N^{ad} . Up to now, N^{ad} must be equal to one. The next entry is the pointer to the lines of the `#Site-Of-Perturbation`-group. Because only one adsorbate atom can be calculated, one finds only one number (4) on that line labelled with `IVADSO(NADSO)`. The 4 points to the fourth line of the `#Site-Of-Perturbation`-group. `NADPO` counts the lines in that group. Any line contains (from the left to the right) the (x, y, z)-coordinates of the adsorbate (`RMAPO`), the pointer to the element (same convention as in the `#Begin/End-Of-Substrate`-group), the value of l_k^{\max} and two parameters prescribing a Gaussian-like (charge and decay) electron density for starting the iterations. `FAKZ` and `OCC` are used if one performs atomic calculations in the vacuum without substrate, and in this example they must be equal to one.

`#Begin/End-Of-Unique-Descriptors`: For a given program unit (first number on each line), the descriptors (or time stamps) prescribe the data and time of its creation. If the program reads from an unit it checks the time stamp and the information in the heading of any file to be consistent. If one runs this example from scratch then the second line (unit 4) of the time stamps has to be changed to the topic date and time of the Green function calculated in ref. [10].

`#Begin/End-Of-Graphics`: This group is used to define cuts and directions for plotting data after the solution is converged. In this example it is of no importance.

`#Begin/End-Of-List-Of-Elements`: The elements are listed in their pseudopotential parametrization. The given example contains only aluminum and silicon.

`#Begin/End-Of-Element`: The first number is the pointer identifying the chemical element. Then follows the charge of the nuclei. In an additional line the experimental covalent radius `COVRAD` is provided.

`#Begin/End-Of-BHS-Potential-Parameters`: These data are divided into a "local" and

“non-local” part. In the local part one finds Z_k^v , α_k^i , and c_k^i . The non-local part contains the parameters $\beta_k^{i,l}$, $a_k^{i,l}$, and $b_k^{i,l}$, separately for s, p, and d orbitals. Compare the parameters with eq. (3).

#Core-Correction: Number of the points to be read for the approximated core charge (given on a logarithmic mesh). The second number is the spacing of the logarithmic mesh. This is used if the linearization of the exchange-correlation functional for the core and valence electrons is not a good approximation (so for Na).

The file `sibasis.d` is printed at the end of this paper as Test Run Input.

References

- [1] P. Hohenberg and W. Kohn, Phys. Rev. 136 (1964) 864.
- [2] W. Kohn and L.J. Sham, Phys. Rev. 140 (1965) 1133.
- [3] D.M. Ceperley and B.J. Alder, Phys. Rev. Lett. 45 (1980) 566.
- [4] J.P. Perdew and A. Zunger, Phys. Rev. B 23 (1981) 5048.
- [5] G.B. Bachelet, D.R. Hamann and M. Schlüter, Phys. Rev. B 26 (1982) 4199.
- [6] M. Scheffler, Ch. Droste, A. Fleszar, F. Máca, G. Wachutka and G. Barzel, Physica B 172 (1991) 143.
- [7] J. Bormet, J. Neugebauer and M. Scheffler, Phys. Rev. B, submitted.
- [8] L.J. Yang, J. Bormet and M. Scheffler, in preparation.
- [9] B. Wenzien, J. Bormet and M. Scheffler, Green function for crystal surfaces I, Comput. Phys. Commun., submitted.
- [10] B. Wenzien, J. Bormet and M. Scheffler, Green function for crystal surfaces II: Projection onto Gaussian orbitals, to be published.
- [11] Ch. Droste, Ph. D. thesis, Fachbereich 4 (Physik) der Technischen Universität Berlin (1990).

TEST RUN INPUT

File sibasis.d

```

#Begin-Of-Iter-Data-Set
#Version-Of-Data-Set
2
#Symmetry
3          * (0): any symmetry, (1) fcc(001), (3) fcc(111)
#Begin-Of-Gaussians * surface adapted coordinate system is used
#Dimension
108, 12, 9 * NDIM, NDEF, NBUFF
#Begin-Of-Sites * Orbital sites in GF (x,y,z), decay, ang. mom.
0.0D0, 0.0D0, -0.577350269189626D0, 0.15D0, 2
0.0D0, 0.0D0, -0.577350269189626D0, 0.38D0, 2
0.0D0, 0.0D0, -0.577350269189626D0, 0.60D0, 2
-0.353553390593274D0, 0.204124145231931D0, 0.0D0, 0.15D0, 2
-0.353553390593274D0, 0.204124145231931D0, 0.0D0, 0.38D0, 2
-0.353553390593274D0, 0.204124145231931D0, 0.0D0, 0.60D0, 2
0.0D0, -0.408248290463863D0, 0.0D0, 0.15D0, 2
0.0D0, -0.408248290463863D0, 0.0D0, 0.38D0, 2
0.0D0, -0.408248290463863D0, 0.0D0, 0.60D0, 2
0.353553390593274D0, 0.204124145231931D0, 0.0D0, 0.15D0, 2
0.353553390593274D0, 0.204124145231931D0, 0.0D0, 0.38D0, 2
0.353553390593274D0, 0.204124145231931D0, 0.0D0, 0.60D0, 2
#End-Of-Sites
#End-Of-Gaussians
#Begin-Of-Box * (origin is the same as for gaussian orbitals)
#Octant
20 * NOCT, points in octant
#Box * (x,y,z)
1.0D0, 0.0D0, 0.0D0 * Box RMRBOX(., 1)
0.0D0, 1.0D0, 0.0D0 * Box RMRBOX(., 2)
0.0D0, 0.0D0, 1.0D0 * Box RMRBOX(., 3)
#Span
21.6939363648002D0 * SPN [bohr]
#End-Of-Box
#Begin-Of-Substrate * (same symmetry as gaussians)
#Type-Of-Substrate * type, local treatment (s=0)
13, 2 * Aluminum
#Lattice-Constant
7.515D0 * Lattice Constant of substrate [bohr]
#Fermi-Energy * relative to Muffin-Tin-Level [eV]
8.800D0 * EFERMI
#Muffin-Tin-Level * [eV]
-3.07444771D0 * VMT
#Work-Function * [eV]
4.3D0 * WORFU
#Geometric-Surface
-0.288675134594813D0 * ZZERO
#Buffer-For-Green-Function
1 * NGFBUF
#End-Of-Substrate
#Begin-Of-Perturbations
#Number-Of-Adsorbates
1 * NADSO
#Pointer-To-Perturbation
4 * IVADSO(NADSO)
#Number-Of-Perturbations
5 * NADPO
#Site-Of-Perturbation * RMADPO(., 1), type, local, start (q, d)
1, 0.00D0, 0.00D0, -0.577350269189626D0, 14, 2, 3.10D0, 0.25D0
2, 0.00D0, 0.00D0, -0.533D0, 14, 2, 3.10D0, 0.25D0

```

```

3, 0.00D0, 0.00D0, -0.4886D0,      14, 2, 3.10D0, 0.25D0
4, 0.00D0, 0.00D0, -0.4540D0,      14, 2, 3.10D0, 0.25D0
5, 0.00D0, 0.00D0, -0.4194D0,      14, 2, 3.10D0, 0.25D0
#Scaled-Perturbation
1.0D0 * FAKZ
#Scaled-Delta-Charge
1.0D0 * OCC
#End-Of-Perturbations
#Begin-Of-Unique-Descriptors
#Number-Of-Descriptors
7
#Descriptors * logical unit, date, time. (integers)
3, 19901211, 101011 * gr (n-contour)
4, 19901211, 1011 * gr (iteration-contour)
9, 19920727, 212638 * is
11, 19920727, 172351 * si
24, 19920804, 161240 * cr
96, 19920804, 160932 * cd
40, 19920203, 200000 * ii
#End-Of-Unique-Descriptors
#Begin-Of-Graphics
#Number-Of-Steps
25, 24 * NPLAN and NRAD
#Step-Range
1.44337567297406D0, 1.22111186D0 * FDELTP and FDELTR
#Number-Of-Drawings
3, 1, 1, 1 * NCUT, NDIR, NWFCT, NEDEN
#Planes * a1, a2, a3, b1, b2, b3, Offset: o1, o2, o3 (x,y,z)
-1.0D0,0.0D0,0.0D0, 0.0D0,0.0D0,-1.0D0, 0.0D0,0.0D0,0.0D0
0.0D0,-1.0D0,0.0D0, 0.0D0,0.0D0,-1.0D0, 0.0D0,0.0D0,0.0D0
-1.0D0,0.0D0,0.0D0, 0.0D0,-1.0D0,0.0D0, 0.0D0,0.0D0,0.0D0
#Directions * a1, a2, a3, Offset: o1, o2, o3 (x,y,z)
0.0D0,-0.408248290463863D0,0.454D0, 0.0D0,0.0D0,-0.454D0
#Energy-Of-Wave-Function * in eV, EWFACT, NSPWF
8.8D0, 3
#Energy-Range-For-Charge * in eV, Emin, Emax
7.0D0, 8.0D0
#End-Of-Graphics
#Begin-Of-List-Of-Elements
#Number-Of-Elements
2
#Begin-Of-Element
#Type-Of-Element * type, charge of nuclei
14, 14.0D0 * Si
#Covalent-Radius * Si experimental
2.098D0 * COVRAD [bohr]
#Begin-Of-BHS-Potential-Parameters * Si of FHI
#Local
4.0D0 * Valency
2.16D0, 0.86D0, 1.6044D0, -0.6044D0 * decays, constants
* d1, d2, c1, c2
#Non-Local
2.40D0, 2.67D0, 3.32D0
-31282.5029298036D0, 26617.4089692391D0, 4675.45239398890D0
2983.68842102636D0, 7527.63343817730D0, 990.299717554505D0
1.24D0, 1.61D0, 2.10D0
-155.801764454364D0, -42.6981043550161D0, 204.130342766202D0
14.6381060580222D0, 105.550700472344D0, 39.1133073194108D0
1.62D0, 1.83D0, 1.94D0
220888.470870308D0, 1525507.68136707D0, -1746392.24511167D0
-12593.8763319792D0, -150078.414567912D0, -75820.6826720308D0
* decays 1, 2, 3 \
* a 1, 2, 3 - three times for l=0,1,2
* b 1, 2, 3 /
* Vi(r) = (ai + r*r*bi) * exp(-di*r*r)

```



```

#End-Of-BHS-Potential-Parameters
#Core-Correction * number of radial mesh-points, log. mesh
0 0.0D0
#End-Of-Element
#Begin-Of-Element
#Type-Of-Element * type, charge of nuclei
13, 13.0D0 * Al
#Covalent-Radius * Al experimental
2.23D0 * COVRAD [bohr]
#Begin-Of-BHS-Potential-Parameters * Al of FHI
#Local
3.0D0 * Valency
1.77D0, 0.70D0, 1.7886D0, -0.7886D0 * decays, constants
* d1, d2, c1, c2
#Non-Local
1.85D0, 2.17D0, 2.34D0
-21399.2454461336D0, -149130.398387931D0, 170537.043024465D0
1833.58400227064D0, 22531.5541572220D0, 11484.3848056219D0
0.93D0, 1.09D0, 1.37D0
-1254.71433301360D0, 740.969038332463D0, 517.890749424891D0
63.1318145427416D0, 231.970961290708D0, 50.7266557261341D0
1.37D0, 1.58D0, 1.75D0
60308.8537018445D0, 106243.207947003D0, -166549.260723623D0
-3575.24090697685D0, -26650.1036827611D0, -10755.3964881995D0
* decays 1, 2, 3 \
* a 1, 2, 3 - three times for l=0,1,2
* b 1, 2, 3 /
* Vi(r) = (ai + r*r*bi) * exp(-di*r*r)
#End-Of-BHS-Potential-Parameters
#Core-Correction * number of radial mesh-points, log. mesh
0 0.0D0
#End-Of-Element
#End-Of-List-Of-Elements
#End-Of-Iter-Data-Set

```

TEST RUN OUTPUT

```

*** INFO ID 1 *** MODULE INIERR: ERROR HANDLING INITIALIZED.
*** INFO ID 1 *** MODULE ITER: BEGIN OF PROGRAM.
*** INFO ID 1 *** MODULE ITER: BASIC DIMENSIONS I. MDIM = 108. MOCT
*** = 20.
*** INFO ID 1 *** MODULE ITER: BASIC DIMENSIONS II. MBUFF = 9.
*** MGFBUFF = 79.
*** NOTE ID 27 *** MODULE RDBADI: CHOICE OF SYMMETRY: (0): NO SYMMETRY,
*** (1): FCC(001), (3): FCC(111). IFCC = 3.
*** INFO ID 1 *** MODULE RDBADI: DIMENSIONS. NDIM = 108. NDEF =
*** 12.
*** INFO ID 1 *** MODULE RDBADI: RS BUFFER. NBUFF = 9.
*** INFO ID 1 *** MODULE RDBADI: DIMENSION OF OCTANT. NOCT = 20.
*** BOX: 1.0000E+00 0.0000E+00 0.0000E+00
*** 0.0000E+00 1.0000E+00 0.0000E+00
*** 0.0000E+00 0.0000E+00 1.0000E+00
*** INFO ID 14 *** MODULE RDBADI: SPAN OF BOX IN A. U. SPN =
*** 2.1693936364800E+01.
*** INFO ID 14 *** MODULE RDBADI: LATTICE CONSTANT IN A. U. SPA =
*** 7.5150000000000E+00.
*** INFO ID 16 *** MODULE RDBADI: EFERMI IN EV. EFERMI =
*** 8.8000000000000E+00.
*** INFO ID 16 *** MODULE RDBADI: MUFFIN-TIN LEVEL IN EV. VMT =
*** -3.0744477100000E+00.
*** INFO ID 16 *** MODULE RDBADI: WORK FUNCTION IN EV. WORFU =
*** 4.3000000000000E+00.
*** INFO ID 1 *** MODULE RDBADI: GEOMETRIC SURFACE. ZZERO =
*** -2.8867513459481E-01.

```

```

*** INFO ID 1 *** MODULE DIVERG: ION-ION FILE READ, ORIGINATING FROM DATE
*** = 19920203. TIME = 200000.
*** INFO ID 1 *** MODULE ITER: LOOP-PARAMETER SET.
*** INFO ID 1 *** MODULE PVMINI: NO PVM USAGE.
*** INFO ID 1 *** MODULE ITER: SCREENING CONSTANTS. C1 =
*** 1.6044000000000E+00. C2 = -6.0440000000000E-01.
*** INFO ID 1 *** MODULE ITER: SCREENING DECAYS. A1 =
*** 2.1600000000000E+00. A2 = 8.6000000000000E-01.
*** INFO ID 1 *** MODULE ITER: GAUSSIAN ENERGY. VGAUS =
*** -7.8624810967933E-01. VGAUSN = -7.8640062344029E-01.
*** INFO ID 1 *** MODULE VVHARX: CENTER OF GRAVITY. ZCG =
*** -1.4553737658350E+00.
*** INFO ID 1 *** MODULE VVHARX: DIPOLE MOMENT. CHARGE =
*** 2.5076583174180E-05. COG = 7.8257449366000E+00.
*** INFO ID 1 *** MODULE VVHARX: AVERAGE CHARGE BEFORE FFT. DT =
*** -1.6671363994814E-07.
*** INFO ID 1 *** MODULE VVHARX: OTH FOURIERCOEFFICIENT OF POTENTIAL REAL
*** = -4.1309124697007E-09. IMAG = 0.0000000000000E+00.
*** INFO ID 1 *** MODULE STEVXC: ZERO VALUES OF CHARGE DENSITY. IP =
*** 19837.
*** INFO ID 2 *** MODULE STEVXC: SMALLEST VALUE OF RHO. RLOW =
*** 2.3797962891805E-36.
*** INFO ID 1 *** MODULE DVEXDR: LOCAL TREATED PART. ILOCAL = 2.
*** FORCE: 1.30709829978E-011 7.29023508254E-004 7.18940569296E-001
*** INFO ID 1 *** MODULE ITER: ITERATION. IT = 1.
*** INFO ID 1 *** MODULE ITER: CHARGE. ORBITALS = 1.0856730398416E+01.
*** BOX = 1.0856730454969E+01.
*** INFO ID 1 *** MODULE ITER: CONT. PHASE = 1.1614876177755E+01. NORM =
*** 6.4017906376969E+00.
*** INFO ID 1 *** MODULE ITER: CONT. MIX = 9.7000000000000E-01. TIME =
*** 4.7060000000000E+02.
*** INFO ID 1 *** MODULE ITER: CONT. ETOT = 0.0000000000000E+00. HARP =
*** 0.0000000000000E+00.
*** INFO ID 1 *** MODULE VVHARX: CENTER OF GRAVITY. ZCG =
*** -1.8843045052129E+00.
*** INFO ID 1 *** MODULE VVHARX: DIPOLE MOMENT. CHARGE =
*** -2.6775474483846E-06. COG = 6.1100219791483E+00.
*** INFO ID 1 *** MODULE VVHARX: AVERAGE CHARGE BEFORE FFT. DT =
*** -8.9924155038517E-08.
*** INFO ID 1 *** MODULE VVHARX: OTH FOURIERCOEFFICIENT OF POTENTIAL REAL
*** = -2.2281851390898E-09. IMAG = 0.0000000000000E+00.
*** INFO ID 1 *** MODULE STEVXC: ZERO VALUES OF CHARGE DENSITY. IP =
*** 20036.
*** INFO ID 2 *** MODULE STEVXC: SMALLEST VALUE OF RHO. RLOW =
*** 2.0621612997937E-36.
*** INFO ID 1 *** MODULE DVEXDR: LOCAL TREATED PART. ILOCAL = 2.
*** FORCE: 3.85901191420E-010 2.22317934072E-004 3.84308617215E-001
*** INFO ID 1 *** MODULE ITER: ITERATION. IT = 2.
*** INFO ID 1 *** MODULE ITER: CHARGE. ORBITALS = 7.3656821912318E+00.
*** BOX = 7.3656822874644E+00.
*** INFO ID 1 *** MODULE ITER: CONT. PHASE = 7.7928989204284E+00. NORM =
*** 3.6600805684710E+00.
*** INFO ID 1 *** MODULE ITER: CONT. MIX = 9.7000000000000E-01. TIME =
*** 4.6916000000000E+02.
*** INFO ID 1 *** MODULE ITER: CONT. ETOT = -9.5614115180958E+01. HARR =
*** -1.0060818063517E+02.
*** INFO ID 1 *** MODULE VVHARX: CENTER OF GRAVITY. ZCG =
*** -2.4845419795120E+00.
*** INFO ID 1 *** MODULE VVHARX: DIPOLE MOMENT. CHARGE =
*** 5.5503014170617E-06. COG = 3.7090720819519E+00.
*** INFO ID 1 *** MODULE VVHARX: AVERAGE CHARGE BEFORE FFT. DT =
*** -4.2601633426313E-08.
*** INFO ID 1 *** MODULE VVHARX: OTH FOURIERCOEFFICIENT OF POTENTIAL REAL
*** = -1.0556042299554E-09. IMAG = 0.0000000000000E+00.
*** INFO ID 1 *** MODULE STEVXC: ZERO VALUES OF CHARGE DENSITY. IP =
*** 20142.
*** INFO ID 2 *** MODULE STEVXC: SMALLEST VALUE OF RHO. RLOW =
*** 1.8930441316306E-36.

```

```

*** INFO ID 1 *** MODULE DVEXDR: LOCAL TREATED PART. ILOCAL = 2.
*** FORCE: 2.87492145113E-010 -2.98572964220E-005 1.76412607808E-001
*** INFO ID 1 *** MODULE ITER: ITERATION. IT = 3.
*** INFO ID 1 *** MODULE ITER: CHARGE. ORBITALS = 5.5827751234844E+00.
*** BOX = 5.5827752460461E+00.
*** INFO ID 1 *** MODULE ITER: CONT. PHASE = 5.8755864290122E+00. NORM =
*** 1.9803645241134E+00.
*** INFO ID 1 *** MODULE ITER: CONT. MIX = 9.7000000000000E-01. TIME =
*** 4.6959000000000E+02.
*** INFO ID 1 *** MODULE ITER: CONT. ETOT = -4.4694508362549E+01. HARR =
*** -4.5921723253045E+01.
*** INFO ID 1 *** MODULE VVHARX: CENTER OF GRAVITY. ZCG =
*** -3.0419594342496E+00.
*** INFO ID 1 *** MODULE VVHARX: DIPOLE MOMENT. CHARGE =
*** 5.4475924475838E-06. COG = 1.4794022630014E+00.
*** INFO ID 1 *** MODULE VVHARX: AVERAGE CHARGE BEFORE FFT. DT =
*** -1.9033608411725E-08.
*** INFO ID 1 *** MODULE VVHARX: OTH FOURIERCOEFFICIENT OF POTENTIAL REAL
*** = -4.7162410129486E-10. IMAG = 0.0000000000000E+00.
*** INFO ID 1 *** MODULE STEVXC: ZERO VALUES OF CHARGE DENSITY. IP =
*** 20200.
*** INFO ID 2 *** MODULE STEVXC: SMALLEST VALUE OF RHO. RLOW =
*** 1.8204401460578E-36.
*** INFO ID 1 *** MODULE DVEXDR: LOCAL TREATED PART. ILOCAL = 2.
*** FORCE: 1.52293055306E-010 -1.25810508480E-004 6.44358274194E-002
*** INFO ID 1 *** MODULE ITER: ITERATION. IT = 4.
*** INFO ID 1 *** MODULE ITER: CHARGE. ORBITALS = 4.6944709348687E+00.
*** BOX = 4.6944710749501E+00.
*** INFO ID 1 *** MODULE ITER: CONT. PHASE = 4.9241266998301E+00. NORM =
*** 1.0383738856192E+00.
*** INFO ID 1 *** MODULE ITER: CONT. MIX = 9.7000000000000E-01. TIME =
*** 4.6952000000000E+02.
*** INFO ID 1 *** MODULE ITER: CONT. ETOT = -2.2015703809829E+01. HARR =
*** -2.2329524867725E+01.
*** INFO ID 1 *** MODULE VVHARX: CENTER OF GRAVITY. ZCG =
*** -3.4328901972876E+00.
*** INFO ID 1 *** MODULE VVHARX: DIPOLE MOMENT. CHARGE =
*** -6.3517936758806E-06. COG = -8.4320789150397E-02.
*** INFO ID 1 *** MODULE VVHARX: AVERAGE CHARGE BEFORE FFT. DT =
*** -8.2337163701517E-09.
*** INFO ID 1 *** MODULE VVHARX: OTH FOURIERCOEFFICIENT OF POTENTIAL REAL
*** = -2.0401897296619E-10. IMAG = 0.0000000000000E+00.
*** INFO ID 1 *** MODULE STEVXC: ZERO VALUES OF CHARGE DENSITY. IP =
*** 20230.
*** INFO ID 2 *** MODULE STEVXC: SMALLEST VALUE OF RHO. RLOW =
*** 1.7916865828791E-36.
*** INFO ID 1 *** MODULE DVEXDR: LOCAL TREATED PART. ILOCAL = 2.
*** FORCE: 9.77414248289E-011 -1.38605810693E-004 8.33261246528E-003
*** INFO ID 1 *** MODULE ITER: ITERATION. IT = 5.
*** INFO ID 1 *** MODULE ITER: CHARGE. ORBITALS = 3.5487714403910E+00.
*** BOX = 3.5487716141596E+00.
*** INFO ID 1 *** MODULE ITER: CONT. PHASE = 3.7006084328970E+00. NORM =
*** 5.4932987896083E-01.
*** INFO ID 1 *** MODULE ITER: CONT. MIX = 9.2150000000000E-01. TIME =
*** 4.6983000000000E+02.
*** INFO ID 1 *** MODULE ITER: CONT. ETOT = -1.1191444077290E+01. HARR =
*** -1.1266940842639E+01.
.....
*** INFO ID 1 *** MODULE WROS: CHARACTERISTIC SUMS FOR CMVO. SUM =
*** -3.0103543509492E+02. ABS = 8.3919391322640E+03.
*** INFO ID 11 *** MODULE ZSYMCL: CMVO IS HERMITIAN. IEPS = 8.
*** INFO ID 1 *** MODULE WROS: CHARACTERISTIC SUMS FOR CMCD. SUM =
*** 2.2997573228048E+00. ABS = 5.6780570069409E+02.
*** INFO ID 1 *** MODULE WROS: CHARACTERISTIC SUMS FOR RMRR. SUM =
*** 2.2245539305321E+01. ABS = 7.1684653983886E-01.
*** INFO ID 1 *** MODULE WROS: SOLUTION WRITTEN.
*** INFO ID 1 *** MODULE ITER: REGULAR END OF PROGRAM AFTER ITERATION
*** CYCLE.

```

STOP