Appendix A

Abbreviations

AO	atomic orbital
AES	Auger emission spectroscopy
AFM	atomic force microscopy
BIS	bremsstrahlung isochromat spectroscopy
BLAS	basic linear algebra subroutines
COHSEX	Coulomb-hole / screened exchange
CPU	central processing unit
DFT	density functional theory
DOS	density of states
EXX	exact exchange
FET	field effect transistor
FFT	fast Fourier transform
GGA	generalised gradient approximation
GW	is *no* abbreviation, but stands for the product of Green's function G and screened interaction W
GWA	GW approximation
ELS-LEED	energy-loss spectroscopy of low-energy electron diffraction
HEG	homogeneous electron gas
HREELS	high resolution electron energy loss spectroscopy
IPES	inverse photoemission spectroscopy
IR	infrared
IRAS	infrared reflection absorption spectroscopy
KS	Kohn-Sham

LAPW	linear augmented plane waves
LCAO	linear combination of atomic orbitals
LDOS	local density of states
LDA	local-density approximation
LEED	low-energy electron diffraction
LEIS	low-energy ion scattering
MBPT	many-body perturbation theory
MIES	metastable impact electron spectroscopy
ML	monolayer
OEP	optimised effective potential
PES	photoemission spectroscopy
PRE	parallel repetition error
RPA	random phase approximation
SCF	self-consistent field
SXRD	surface X-ray diffraction
STM	scanning tunnelling microscopy
UPS	ultraviolet photoelectron spectroscopy
UV-VIS	ultraviolet & visible [light]
XPS	X-ray photoelectron spectroscopy
XRD	X-ray diffraction

Appendix B

Dielectric models

B.1 Image-charge method for dielectric layer models

In this section a simple scheme is presented to compute the screened interaction in laterally homogeneous model systems with a layered structure (cf. Fig. B.1a). Each layer z has the same thickness L and a layer-specific dielectric constant ε_z. To simplify the notation, we work with reduced units in the following, i.e. lengths are measured in units of L, charges in units of the unit charge Q, and potentials in units of Q/L. Our coordinate system is chosen such that the layers are centred around integer L, and the interfaces are at half integer L. We note that any given dielectric profile $\varepsilon(z)$ can be approximated by such a layer model when the profile is discretised into individual layers of a sufficiently small thickness. For the model calculations in this work, we usually use L=1 bohr. At the boundary between two layers we assume sharp interfaces so that the dielectric constant jumps from ε_z to ε_{z+1}.

The screened interaction $W(\mathbf{r}, \mathbf{r}')$ is obtained as the potential $V(\mathbf{r})$ when a unit charge is placed at $\mathbf{r}'$. We compute $V(\mathbf{r})$ by the method of image charges [90]. As an introductory remark, let us first consider the textbook situation of two semi-infinite dielectric media Ω_A and Ω_B with dielectric constants ε_A and ε_B, sketched in Fig. B.1b. For a charge q at $\mathbf{r}'$ in Ω_A, the potential $V(\mathbf{r})$ is given by

$$\mathbf{r} \in \Omega_A: \quad V(\mathbf{r}) = \frac{1}{\varepsilon_A}\left(\frac{q}{|\mathbf{r}-\mathbf{r}'|} + \frac{q''}{|\mathbf{r}-\mathbf{r}''|}\right), \tag{B.1}$$

$$\mathbf{r} \in \Omega_B: \quad V(\mathbf{r}) = \frac{1}{\varepsilon_B}\frac{q'}{|\mathbf{r}-\mathbf{r}'|}, \tag{B.2}$$

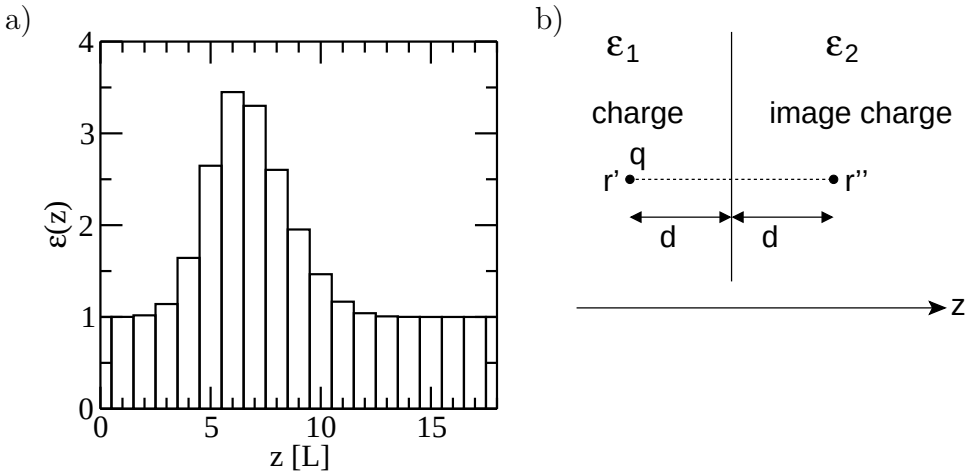

Figure B.1: a) Dielectric layer model. b) Charge and image charge at a dielectric interface.

where q' and q'' are image charges. They are determined from the continuity equations of the electric field and the electric displacement at the interface, yielding

$$q' = \frac{2\varepsilon_B}{\varepsilon_A + \varepsilon_B}\, q \qquad \text{(B.3)} \qquad\qquad q'' = \frac{\varepsilon_A - \varepsilon_B}{\varepsilon_A + \varepsilon_B}\, q\ . \qquad \text{(B.4)}$$

$\mathbf{r}''$ is obtained by reflecting $\mathbf{r}'$ at the interface, and we will therefore denote q'' as "reflected charge", whereas the effect of q is propagated into Ω_B by the "propagated charge" q'. The image charges q' and q'' are no physically observable charges, but only mathematical constructs to simplify the computation of the potential. Most importantly, the image charges for the potential on one side of the interface are always located on the other side of the interface, whereas the original charge is on the same side. This criterion is useful in multi-layer systems to identify the interfaces for higher-order image charges.

In order to develop a computational scheme for a multi-layer system, a proper book-keeping is crucial to keep track of the various image charges. For reasons that will become clear below, we denote an image charge that contributes to the potential in layer z and is located at $z + \sigma d$ by $q(z, d, \sigma)$, where $d \geq 0$ is the distance from the layer and $\sigma = \pm 1$. To show that the image charges can be determined iteratively we will now derive the iteration for $d \to d + 1$. Consider a charge $q(z, d, \sigma)$ relevant for the potential in layer z. Due to the interface at $z - \frac{1}{2}\sigma$ two additional image charges appear. The

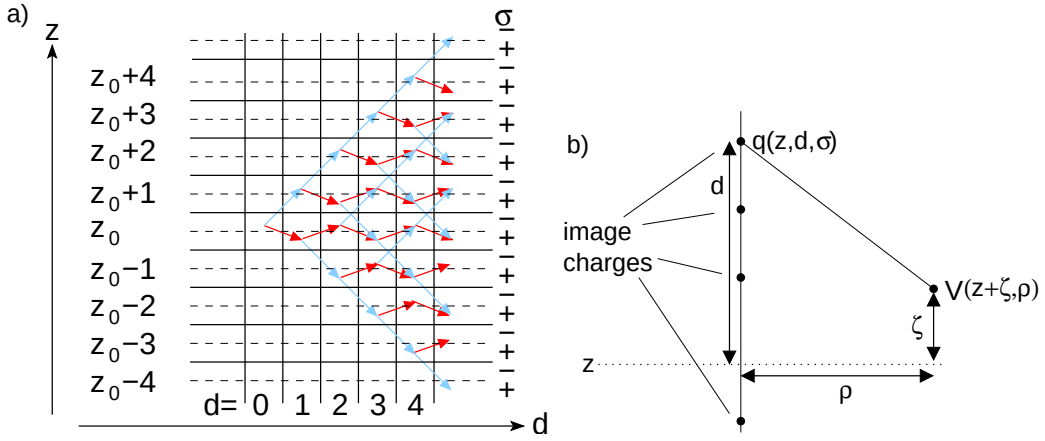

Figure B.2: a) Iterative determination of image charges (see text). b) Computation of the potential from the image charges.

reflected image charge, located at

$$\underbrace{z-\frac{\sigma}{2}}_{\text{interface}} \; - \; \underbrace{((z+\sigma d)-(z-\frac{\sigma}{2}))}_{\text{distance from interface}} \; = z-\sigma(d+1) \tag{B.5}$$

describes the potential in layer z and is given by (cf. Eq. B.4)

$$q^{\mathrm{rf}}(z,d+1,-\sigma) = \frac{\varepsilon_z-\varepsilon_{z-\sigma}}{\varepsilon_z+\varepsilon_{z-\sigma}}\, q(z,d,\sigma)\ . \tag{B.6}$$

The propagated image charge remains at the position $z+\sigma d$ and describes the potential in layer $z-\sigma$. Using our book-keeping notation and Eq. B.3 it can be written as

$$q^{\mathrm{pr}}(z-\sigma,d+1,\sigma) = \frac{2\varepsilon_{z-\sigma}}{\varepsilon_z+\varepsilon_{z-\sigma}}\, q(z,d,\sigma)\ . \tag{B.7}$$

Obviously, the distance parameter is increased by 1 for each interface taken into account. The image charges for the distance $d+1$ can thus be computed iteratively from those at distance d and will in general combine a reflected and a propagated contribution $q = q^{\mathrm{rf}} + q^{\mathrm{pr}}$.

The dielectric discontinuity at the interfaces between two layers introduces a divergence of the potential close to the interface. However, this is an artefact of assuming homogeneous layers with sharp interfaces. To avoid it, we restrict the position of the original charge to the center of each slab; the image charges are then located at the center of a layer, too. The iterations are then started by setting

$$q(z_0,0,\pm 1) = 1\ . \tag{B.8}$$

The flow of the iterations is schematically depicted in Fig. B.2. Each rectangular box in the scheme corresponds to one image charge $q(z, d, \sigma)$ and the red (light blue) arrows indicate the reflected (propagated) contributions to the image charges of the next generation. For clarity, only the flow for $q(z_0, 0, -1)$ is shown. In practice, the iterations are stopped at some $d_{\max}$ which thereby becomes a convergence parameter. In the iteration scheme in Fig. B.2a this corresponds to stop going to the right. In addition, we truncate the system and neglect image charges that fall outside, which corresponds to ignoring charges at the bottom or the top in Fig. B.2a. This truncation becomes a second convergence parameter. The convergence for both parameters was tested by doubling the parameter until the changes became negligible.

The screened potential depends on the layer z and the lateral distance ρ from the vertically aligned image charges. By summing the Coulomb potential of all the image charges relevant for this layer (cf. Fig. B.2b), we obtain

$$V(z+\zeta, \rho) = \frac{1}{\varepsilon_z} \sum_{d=0}^{d_{\max}} \sum_{\sigma} \frac{q(z, d, \sigma)}{\sqrt{(\zeta - \sigma d)^2 + \rho^2}} \,, \tag{B.9}$$

where ζ denotes the vertical position within the layer ($|\zeta| < \frac{1}{2}$). We usually restrict the calculation to $\zeta = 0$. When Eq. B.9 is evaluated successively during the iterations, there is no need to store the image charges for all d, which makes the implementation very memory-efficient.

B.2 Connection to G_0W_0: the static COHSEX approximation

In this section, we will show how the screened interaction calculated from dielectric models can be used to estimate changes in the quasiparticle energies. In the end, we will arrive at a very simple scissors operator. We will then show that the same result can be obtained by considering the image-potential energy of the charged N±1-electron systems that result from an electronic excitation.

The G_0W_0 self-energy can be decomposed into physically meaningful entities. One such possibility is to decompose it into a screened exchange and a Coulomb-hole part. To this end, we start from the expression for the self-energy in the frequency domain

$$\Sigma(\mathbf{r}, \mathbf{r}', \omega) = \frac{i}{2\pi} \int \mathrm{d}\omega' G_0(\mathrm{r}, \mathrm{r}', \omega + \omega') W_0(\mathrm{r}, \mathrm{r}', \omega') e^{i0^+\omega'} \,. \tag{B.10}$$

The residual theorem states that only the poles of G_0 and W_0 contribute to the integral. Closing the integration contour above the real axis includes only the poles of the Green's function for the occupied states [23], which leads to the screened exchange self energy

$$\Sigma_{\mathrm{sx}}(\mathbf{r},\mathbf{r}',\omega) = -\sum_{n}^{\mathrm{occ}} \phi_n(\mathbf{r})\phi_n^*(\mathbf{r}')W(\mathbf{r},\mathbf{r}',\omega-\epsilon_n) \; . \tag{B.11}$$

The poles of the screened interaction are given by the plasmon energies $\pm(\omega_p - i0^+)$, i.e.

$$W(\mathbf{r},\mathbf{r}',\omega_p) = v(\mathbf{r},\mathbf{r}') + \sum_p \frac{2\omega_p\chi_p(\mathbf{r})\chi_p^*(\mathbf{r}')}{\omega^2-(\omega_p-i0^+)^2} \; , \tag{B.12}$$

where χ_p denotes the corresponding plasmon functions. When we expand also G_0 in its spectral representation, we arrive at the Coulomb-hole self-energy

$$\Sigma_{\mathrm{coh}}(\mathbf{r},\mathbf{r}',\omega) = \sum_p\sum_n \frac{\phi_n(\mathbf{r})\phi_n^*(\mathbf{r}')}{\omega-\omega_p-\epsilon_n}\chi_p(\mathbf{r})\chi_p^*(\mathbf{r}') \; . \tag{B.13}$$

Assuming a static interaction (or more precisely, $\omega - \epsilon_n \ll \omega_p$ for the frequency range of interest) leads to the static COHSEX approximation. Comparison of the equations for Σ_{coh} and Σ_{sx} to Eq. B.12 for $\omega = 0$ and exploiting the identity

$$\sum_n \phi_n(\mathbf{r})\phi_n^*(\mathbf{r}') = \delta(\mathbf{r}-\mathbf{r}') \tag{B.14}$$

we obtain

$$\Sigma = \Sigma_{\mathrm{coh}} + \Sigma_{\mathrm{sx}} \; , \tag{B.15}$$

$$\Sigma_{\mathrm{coh}}(\mathbf{r},\mathbf{r}') = \frac{1}{2}(W(\mathbf{r},\mathbf{r}) - v(\mathbf{r}-\mathbf{r}'))\delta(\mathbf{r}-\mathbf{r}') \; , \tag{B.16}$$

$$\Sigma_{\mathrm{sx}}(\mathbf{r},\mathbf{r}') = -\sum_n^{\mathrm{occ}} \phi_n(\mathbf{r})\phi_n^*(\mathbf{r}')W(\mathbf{r},\mathbf{r}') \; . \tag{B.17}$$

A particularly useful expression can be derived from the static COHSEX approach when it is applied to long-range screening effects in separated subsystems. For each subsystem, the presence of the other subsystems does not alter the Green's function. However, the polarisation of the other subsystems adds a contribution ΔW^{p} to the screened interaction which varies only smoothly, i.e., is essentially constant over the subsystem as shown in Section 3.3.3. The additional self-energy then becomes

$$\Delta\Sigma^{\mathrm{p}}(\mathbf{r},\mathbf{r}') = \frac{1}{2}\sum_n^{\mathrm{unocc}} \phi_n(\mathrm{r})\phi_n^*(\mathrm{r}')\Delta W^{\mathrm{p}} - \frac{1}{2}\sum_n^{\mathrm{occ}} \phi_n(\mathrm{r})\phi_n^*(\mathrm{r}')\Delta W^{\mathrm{p}} \; . \tag{B.18}$$

This is a symmetric scissors operator that opens the quasiparticle gap by ΔW^{p}.

We arrive at the same result by considering the change in the total energy $E_{N\pm1,s}$ after the electronic excitation. The induced image-potential in the charged final state shifts the energy by $\frac{1}{2}\Delta W$ for both positively charged (hole) states or negatively charged (electron) states. The hole energy is (cf. 2.34)

$$\epsilon_s = E_{N,0} - (E_{N-1,s} + \frac{1}{2}\Delta W) < E^{\mathrm{Fermi}} \tag{B.19}$$

and corresponds to occupied states in the initial system, which are thus shifted by $-\frac{1}{2}\Delta W$. For the electron energy (cf. 2.35)

$$\epsilon_s = (E_{N+1,s} + \frac{1}{2}\Delta W) - E_{N,0} > E^{\mathrm{Fermi}} \tag{B.20}$$

the opposite is true: the final state effect shifts the unoccupied band energy by $+\frac{1}{2}\Delta W$.

Appendix C

Ultrathin oxide films

C.1 Alternative terminations for the α-quartz (0001)

In order to see if the siloxane surface is the most stable surface termination for silica slabs at all thicknesses, we have also investigated unreconstructed quartz slabs with three possible terminations: a Si-terminated surface containing two-fold coordinated silicon atoms at the surface, a O_1-terminated where a single oxygen atom is added to the Si-termination, and an O_2-termination with two oxygen atoms per Si surface atom. Only the O_1-terminated surface yields stoichiometric slabs, whereas the Si-termination (O_2-termination) has a silicon (oxygen) excess. The O_1-terminated surface exhibits an almost planar configuration at the surface silicon atom with a relatively short Si-O bond (1.50 Å). An analysis of the electronic wavefunctions reveals that no strong double-bond character can be detected and that it is better described as a Si^+–O^- entity where the short bond results from a strong Coulomb attraction. We note here that the the Si_4O_8 film with an O_1-termination spontaneously decomposed into two reconstructed Si_2O_4 layers during a standard relaxation. Only when some of the central atoms were kept frozen, the structure analogous to the Si_3O_6 and Si_5O_{10} films could be obtained. This indicates that such under-coordinated Si-O species are highly reactive and may play an important role in the restructuring processes in real materials. Also at the O_2-terminated surface, we observe a reaction: the oxygen atoms dimerise. The resulting structure can be understood as a peroxide ion coordinated side-on to the silicon centre (cf. Fig. C.1). This assignment is supported by the O-O bond length of 1.60 Å characteristic for oxygen single bonds, and by the electronic structure. Similar dimerisations are observed when oxygen atoms are adsorbed on oxide surfaces [148].

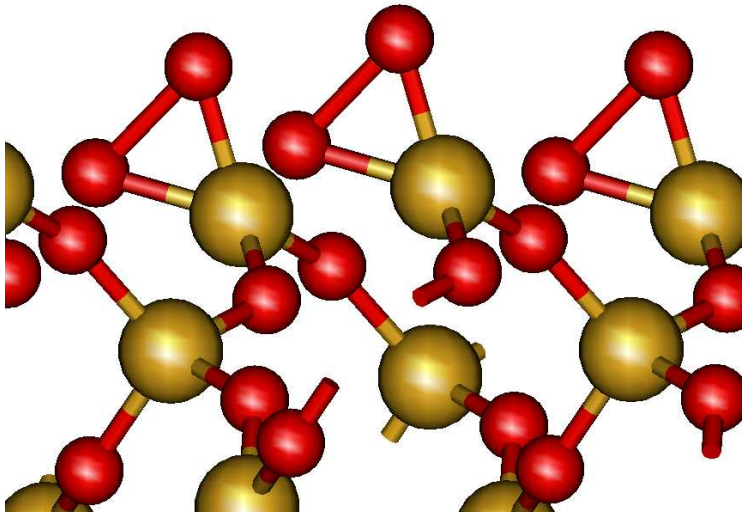

Figure C.1: The relaxed O_2-terminated surface structure with a direct O-O peroxide bond.

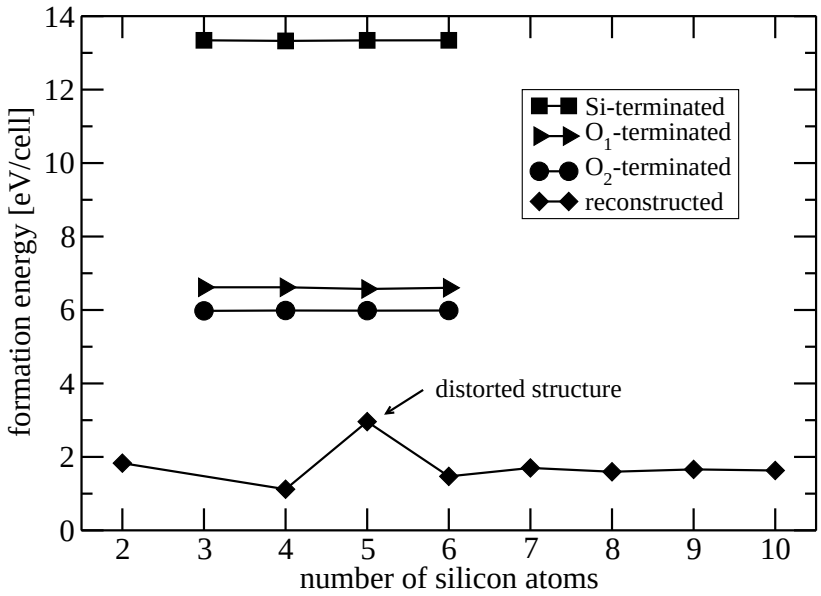

Figure C.2: Formation energies (DFT-LDA) of thin quartz-like slabs with various terminations as well as the reconstructed silica slabs as a function of thickness.

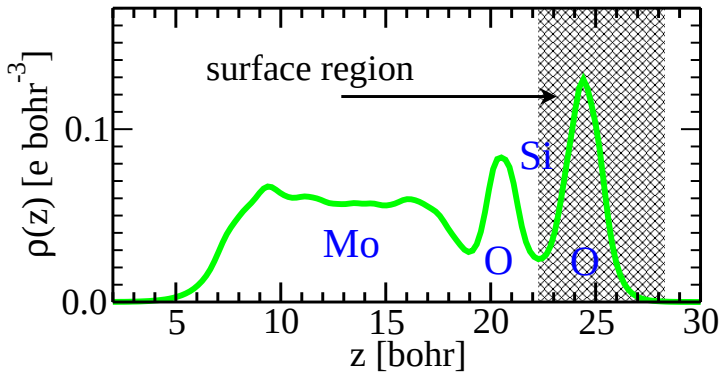

Figure C.3: Definition of the surface region for the $Mo_{10}Si_2O_5$ system.

In Fig. C.2 the formation energies of the slabs from bulk α-quartz and molecular oxygen[1] according to

$$n\ SiO_2 + \frac{m}{2}\ O_2 \longrightarrow Si_nO_{2n+m}$$

are plotted as a function of n. m is the surface oxygen excess, ranging from -2 for the Si-terminated slab to +2 for the O_2-terminated one. The non-reconstructed structures show no visible dependence of the formation energy on the slab thickness. In other words the formation energy is determined by the surface only and reaches a constant value already for very small slabs. The formation energy of the reconstructed slabs on the other hand shows more oscillations. For slabs up to $n = 6$, this reflects the variations in the slab structure since a quartz-like central part is present only from $n = 6$ on. However, the oscillations continue even for thicker slabs. This can be traced back to the misfit of the siloxane surface structure to the quartz substrate. The resulting strain is accommodated in the substrate and decays only slowly with increasing depth. This can also be monitored by the Si-O-Si angle – the most sensitive parameter to structural deformations – which approaches its bulk value of 140° only slowly with increasing depth. The reconstructed films are more stable than the other terminations, which is not surprising since all dangling bonds are saturated after the reconstruction.

C.2 Surface-projected density of states

For a meaningful comparison between different substrates, the DOS must be projected onto the surface region. Here, we will describe how we define the

[1]Spin-polarized O_2 with the theoretical bond of 1.209 Å. To employ the plane-wave code, the molecule is placed in a large simulation box.

surface region and perform the spatial projection. The z-resolved electron density for the (pseudo)valence states

$$\rho(z) = \frac{1}{A} \int dxdy \; \rho(x, y, z) \; , \tag{C.1}$$

where the integral is taken over the surface unit cell (with area A), exhibits minima and maxima corresponding to the ionic layers (cf. Fig. C.3). Since the valence electronic structure of the oxides is dominated by oxygen-derived states, we can use the minima to divide the system into individual oxygen layers. The last oxygen layer then defines the surface region Ω_{srf}. The surface local density of states (LDOS) is obtained as

$$LDOS(E) = \sum_n \delta(E - \epsilon_n) \int_{\Omega_{\mathrm{srf}}} d^3\mathbf{r} \; |\psi_n(\mathbf{r})|^2 \; , \tag{C.2}$$

i.e. the partial density integrated over Ω_{srf} is used to weight the peaks in the DOS.

Appendix D

NaCl films

D.1 Atomic orbital projections

In order to assess the character of a wave function $\psi_{n\mathbf{k}}$ in a crystal, it is often instructive to decompose it into atomic contributions. While this is very natural when atom-centred local orbitals are used as basis set to expand the wavefunction, plane waves are not associated with particular atoms. It is therefore necessary to project the wavefunctions onto an atomic basis set [149]. Here, we employ the atomic pseudo-wavefunctions that are used to define the pseudopotential projectors for this purpose. We denote the atomic orbitals by χ and employ Greek letters μ, ν, ρ for the indices. From these orbitals, Bloch states $\chi_{\mu\mathbf{k}}$ are formed.

However, atomic orbital basis sets are in general non-orthogonal, i.e. the overlap matrix

$$S_{\mu\nu}(\mathbf{k}) = \langle \chi_{\mu\mathbf{k}} | \chi_{\nu\mathbf{k}} \rangle \tag{D.1}$$

is not diagonal. In the following we will employ a notation in analogy to the covariant and contravariant coordinates in non-orthogonal coordinate systems. The inverse overlap matrix is then written $S^{\mu\nu}(\mathbf{k})$, and is defined by

$$\sum_{\rho} S_{\mu\rho}(\mathbf{k}) S^{\rho\nu}(\mathbf{k}) = \delta_{\mu\nu} \ . \tag{D.2}$$

The projections

$$c_{n\mu}(\mathbf{k}) = \langle \chi_{\mu\mathbf{k}} | \psi_{n\mathbf{k}} \rangle \tag{D.3}$$

then differ from the expansion coefficients $c_n^{\mu}(\mathbf{k})$ that generate the crystal wavefunctions via

$$|\psi_{n\mathbf{k}}\rangle = \sum_{\mu} c_n^{\mu}(\mathbf{k}) |\chi_{\mu\mathbf{k}}\rangle \ . \tag{D.4}$$

The expansion coefficients are connected to the projections by the inverse overlap matrix $S^{\mu\nu}$

$$c_n^\mu(\mathbf{k}) = \sum_\nu S^{\mu\nu}(\mathbf{k}) c_{\nu i}(\mathbf{k}) \ . \tag{D.5}$$

We note that the norm of the wave functions is given in the atomic orbital basis by

$$\langle \psi_{n\mathbf{k}} | \psi_{n\mathbf{k}} \rangle = \sum_\mu c_i^\mu(\mathbf{k}) c_{\mu i}(\mathbf{k}) \ . \tag{D.6}$$

We then define the projection onto a subset of the atomic orbital basis (in general orbitals associated with one specific atom or one class of atoms) by restricting the sum in Equation D.6 to this subset.

The advantage of this technique lies in the fact that it is closely related to the Mulliken population analysis for atomic orbital basis sets. For example, summing the projections over occupied states

$$q_\mu = \int d^3\mathbf{k} \sum_i^{\text{occ}} c_i^\mu(\mathbf{k}) c_{\mu i}(\mathbf{k}) \tag{D.7}$$

recovers the Mulliken gross populations q_μ, which may give a qualitative picture of the charge redistribution in the solid.

In general, the atomic orbital basis set is not complete. Even including all atomic orbitals in the sum does not recover the full (plane-wave) norm. This spill-over amounts to typically 0.5–2% for occupied states but may become larger for unoccupied states. Since we use the projection onto atomic orbitals to assess the character of single bands, we do not reorthonormalise the projected bands as suggested in [149] for the population analysis because this would mix different bands. The spill-over is largely due to the minimal, non-adjusted basis set that results from the atomic pseudo-wavefunctions. We refrain from adapting the atomic orbitals to minimise the spill-over because we are interested in the qualitative picture and not in numbers.

D.2 NaCl: the character of the bulk conduction band

While the valence bands of bulk NaCl are easily understood as being derived from the chlorine $3p$ orbitals, the character of the lowest conduction band has proved to be more difficult to assess and has been a matter of discussion since the earliest investigations by Slater and Shockley [150]. Assigning an atomic character to a delocalised wavefunction is not unique, and different methods yield different results for the conduction band in NaCl. A simple

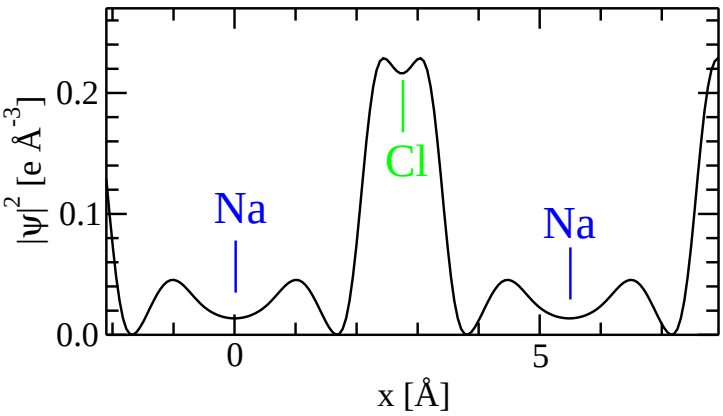

Figure D.1: Partial density profile of the conduction band minimum along the (001) direction, suggesting a dominant Cl character. The position of the Na and Cl atoms is indicated.

linear combination of atomic orbitals (LCAO) picture suggests that the conduction bands should be mainly composed of the sodium states. Indeed, the AO projection technique with the (minimal) atomic pseudo-orbital basis yields 70–93 % Na character for the lowest conduction band. The maximum Na character is obtained for the conduction band minimum. However, the partial density computed from this state is clearly centred on the chlorine atoms. In Fig. D.1 we show the partial density $|\psi|^2$ of the conduction band minimum along the [001] direction through the Na and Cl atoms. The high density close to the Cl nucleus reveals a dominant Cl character for this state. Further evidence for a considerable Cl character has been provided by de-Boer and de Groot [151, 152]: they considered a hypothetical Cl^- fcc lattice with a homogeneous background charge and found that it reproduces the band structure of NaCl very well. Moreover, they showed that the lowest conduction state disappears in a muffin-tin sphere calculation when the Cl 4s orbital is removed from the basis set. They concluded from this evidence that the lowest conduction state has mainly Cl 4s character.

Another way to approach this problem is to monitor how the bands develop when the ions are brought together from infinite separation, i.e. by changing the lattice constant from very large values to its equilibrium value [150]. At large separations, the ions do not interact and the bands reflect the ionic levels.[1] When the ions approach each other, the Madelung potential will lower the anion states (which are surrounded by the positively charged

[1]We note that in this case, the charge transfer from Na to Cl is even endothermic, i.e. the ground state are Na and Cl atoms rather than Na^+ and Cl^-. We will however assume a charge-transfer state at all separations.

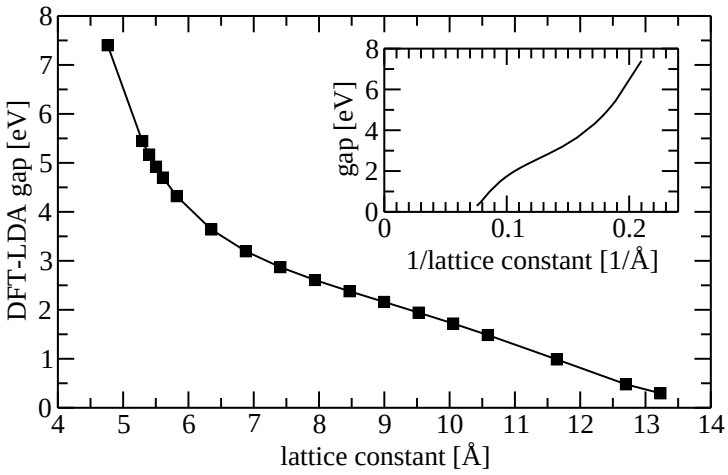

Figure D.2: Dependence of the DFT-LDA gap on the lattice constant for bulk NaCl. The inset shows the same data as function of the inverse lattice constant.

cations) and destabilise the cation states. A second effect is the broadening of the ionic levels into bands when the wavefunctions of the ions overlap. It is then possible that a broadened Cl 4*s* band drops below the corresponding Na 3*s* band at sufficiently small separations, but is this really the case for NaCl?

The change in the band gap should be a sensitive test for this question. With decreasing lattice constant, the sodium states will be shifted upward with respect to the Cl 3*p* states due to the increase in the Madelung potential. A gap for a sodium-derived conduction band should thus increase. On the other hand, the Cl 4s band broadening would increase and lead to a reduction in the band gap if Cl 4s states were dominant in the conduction band. We have therefore computed the band gap over a large range of lattice constants, cf. Fig. D.2. We find that the band gap increases with decreasing ion separation.[2] This indicates an important role of the Na orbitals in determining the band gap.

However, the influence of the Cl ions for the electronic band structure at the equilibrium lattice constant cannot be neglected. In agreement with de-Boer *et al.*, we find that a Cl^- fcc lattice with a neutralising background alone reproduces the conduction band and even the band gap of bulk NaCl, while a corresponding Na^+ lattice shows a free-electron like band structure. Even when the Cl^- ions are modelled by negative point charges for the Na^+-only

[2] At even smaller lattice constants below 4.2 Å, the band gap reduces again since a further band drops below the dispersive band in question.

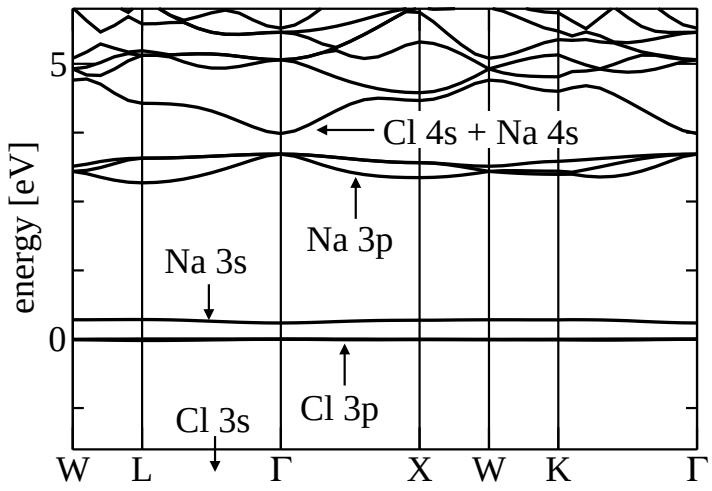

Figure D.3: DFT-LDA band structure of bulk NaCl at a lattice constant of 13.2 Å(25 bohr).

calculation, strong deviations from the NaCl band structure are observed. The Cl ions are absolutely crucial for the observed dispersion of the conduction band. At large lattice constants, however, the Na-derived bands separate from the dispersive band continuum (cf. Fig. D.3) and cross the Cl 3*p* level at ∼14 Å. Surprisingly, we find at a lattice constant of 13.2 Å (25 bohr) that the unoccupied Na 4*s* states hybridise with the chlorine states and produce a dispersion reminiscent of bulk NaCl at the equilibrium lattice constant. None of these aspects can be reproduced with a Cl^- lattice alone. When the lattice constant is varied between the theoretical equilibrium (5.49 Å) and 13.2 Å, the lowest conduction band smoothly transforms into the localised Na 3s band. We thus conclude that the energetic position of the lowest conduction band is coupled to the Na 3*s* state, but that the Na 3*s* state hybridises with a Cl scattering state (that may be denoted as Cl 4*s*). This hybridisation is responsible for the dispersion of the conduction band. That the band structure of NaCl around the equilibrium lattice constant can be reproduced with fcc Cl^- alone can be explained by the fact that the scattering behaviour of the sodium core is similar to that of the vacuum in the relevant energy range. It does hence not mean that the Na would be irrelevant for the conduction states, but that a jellium background is a sufficient approximation for the sodium core in this very case.

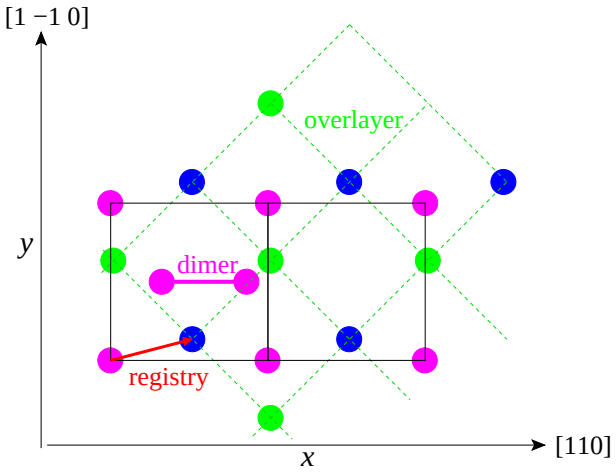

Figure D.4: Schematic representation of the NaCl overlayer on the Ge p(2×1) dimer surface. Pink: Ge, Blue: Na, Green: Cl.

D.3 Structure of the NaCl/Ge(001) interface

To investigate the interface between a sodium chloride overlayer and the Ge(001) $p(2 \times 1)$ surface, a 2 ML NaCl film was put on top of the buckled-dimer Ge(001) surface with various lateral shifts ("registry"). Since there is no experimental evidence that the film thickness influences the interface, the interface structure was not investigated for other thicknesses. To get an overview over the potential energy surface, the lateral position of the top NaCl layer was kept fixed, while the vertical position of this layer was relaxed. The bottom NaCl and the top four Ge layers were relaxed, too. For the lateral position of the top layer ("registry", cf. Fig. D.4), three linescans along the dimers (x-direction) were performed at Δy=0 (Cl above the dimer), 0.25 (low-symmetry), and 0.5 (Na above the dimer).[3] Higher y-offsets need not be considered due to the mirror symmetry of the NaCl overlayer and the substrate, respectively. Likewise, the x-offsets above 0.5 (in relative coordinates of the 2×1 unit cell) have been obtained by symmetry considerations since the (unrelaxed) overlayer has a 1×1 unit cell. We do not observe a dimer flip (down-up $\leftrightarrow$ up-down) during the relaxation.

The linescans are shown in Fig. D.5. We display the adhesion energy of the 2 ML NaCl adlayer as a function of the registry; negative values correspond to binding of the overlayer. It can be clearly seen that the displacement along the dimer plays a very important role. All three linescans have a similar shape and agree for the position of the minima (x=0.1) and maxima

[3] All coordinates are relative with respect to the 2×1 unit cell.

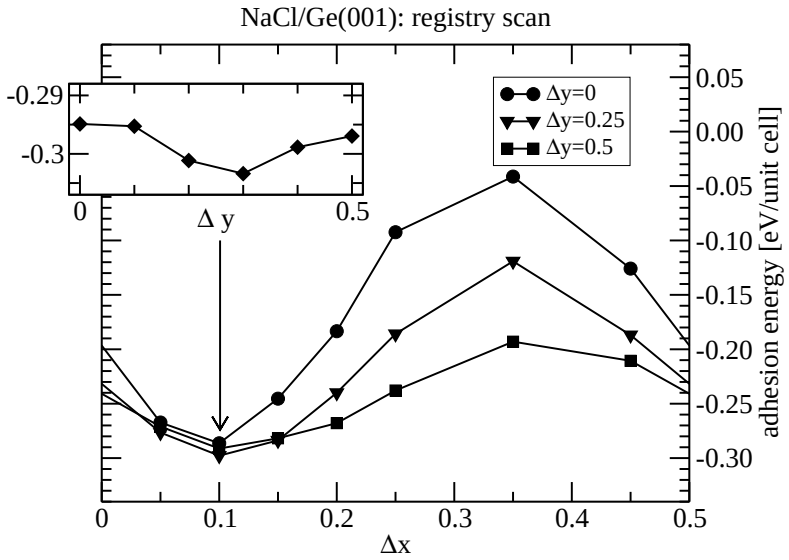

Figure D.5: Linescans for the NaCl registry above the Ge(001) 2×1 surface along the dimers (Δx) and in the perpendicular direction (Δy) close to the minimum.

Δy	Δx	E
0	0.099	-0.310 eV
0.25	0.100	-0.306 eV
0.5	0.100	-0.301 eV

Table D.1: Minimum search for three displacements Δy perpendicular to the dimers (Δy is kept fixed).

(x=0.35). The height of the maximum depends on the displacement along $[1\overline{1}0]$ perpendicular to the dimer axis, whereas the minima are all very close in energy. Therefore, a linescan along $[1\overline{1}0]$ was performed for $\Delta x = 0.1$, which is shown in the inset in Fig. D.5. This linescan demonstrates that the potential energy surface is very flat along this direction. In order to find the minimum, a Δx relaxation with tighter convergence parameters[4] was started from three points Δy=0, 0.25, and 0.5. The result shown in Table D.1 reveals that the minimum is at Δy=0, but is indeed very shallow in the y-direction. The adhesion energy at the minimum of 0.31 eV per unit cell agrees reasonably with the experimental estimate of 0.26 eV [131].

[4]The tighter convergence and the relaxation of the top layer NaCl along x are responsible for the small offset of 5 meV for $\Delta y = 0.5$ between the inset of Fig. D.5 and Tab. D.1.

D.4 STM simulations

According to Tersoff-Hamann approximation [153], the tunnelling current in STM is proportional to the LDOS at the Fermi energy at the position of the tip. To account for the experimental tip bias of 1.5 – 2.7 eV, we integrate the LDOS between the bias U and the Fermi energy. This partial density

$$\rho^{\mathrm{STM}}(\mathbf{r}) = \int_U^0 dE \sum_{n\mathbf{k}} |\psi_{n\mathbf{k}}(\mathbf{r})|^2 \delta(E - \epsilon_{n\mathbf{k}}) \tag{D.8}$$

should then reflect the STM tunnelling currents.[5] We use $U = -2\,\mathrm{eV}$ in the following. A variation of the energy range gives no indication that the STM pictures would change outside the experimental range. The experimental observation that only a small energy range allows for STM pictures cannot be explained from our results. An inspection of ρ^{STM} (not shown) at the surface of the NaCl films reveals that the bright spots in the STM must be assigned to the chloride ions. In conflict with the experimental 1×1 pattern, the theoretical two-layer model invariably produces a 2×1 STM pattern with a chloride corrugation of $\sim 0.3\,\text{Å}$. This result is also independent of the shift of the overlayer perpendicular to the dimers. Positioning the sodium atom above the top dimer atom and the chlorine atom between the dimers removes the geometrical corrugation of 0.2 Å at the surface and changes the total energy by only 10 meV, which may be below the absolute accuracy of our DFT-LDA approach. Nevertheless, the partial density shows a corrugation of $\sim$0.2 Å. We therefore do not believe that this discrepancy between the STM simulations and the experiment indicates an error in the atomistic model. Rather, deficiencies in the STM simulation, e.g. the use of the Tersoff-Hamann approximation or the neglect of atomic relaxation due to the tip-induced electric field may be responsible. The current approximations can hence not account for the observed STM pictures in full detail. However, we find strong evidence that the chlorine atoms appear as bright spots, since the Ge states hybridise only with these over the full valence energy range.

On the other hand, we can reproduce the apparent heights of the NaCl layers in the STM experiments. For this, we average ρ^{STM} laterally and plot it on a logarithmic scale (cf. Fig. D.6 bottom). Above the surface the density decays exponentially. The (local) decay constant $\alpha(z)$ can be obtained via

$$\alpha(z) = \frac{\frac{\mathrm{d}}{\mathrm{d}z}\rho^{\mathrm{STM}}(z)}{\rho^{\mathrm{STM}}(z)} = \frac{\mathrm{d}}{\mathrm{d}z} \ln \rho^{\mathrm{STM}}(z) \tag{D.9}$$

[5] It is possible to introduce an energy-dependent weighting factor in the above integral, containing additional parameters. However, by varying U we observe that the main characteristics of the ρ^{STM} are not sensitive to the energy-dependence and we estimate that an energy-dependent weighting would not introduce qualitative differences.

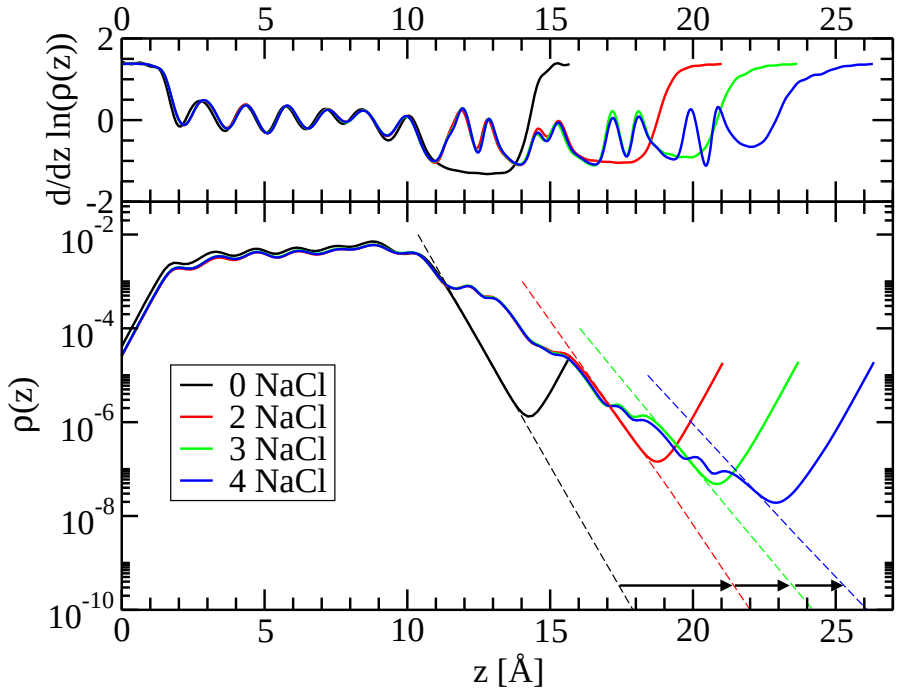

Figure D.6: Laterally averaged partial densities for the STM simulations for a 6 layer Ge slab with a NaCl of 2,3, and 4 layer thickness and without a NaCl layer (0 NaCl). The top figure shows the local decay constants. The dashed lines indicate the exponential decay, and the distance between the curves correspond to the apparent heights in STM. The density increase is an artifact due to the periodicity in the simulations.

and is shown in the top graph of Fig. D.6. When NaCl layers are adsorbed on the Ge substrate, the average density curves are shifted towards higher z as indicated by the black arrows in Fig. D.6. When the STM is used in constant current mode, an equidensity surface is scanned. We can therefore identify the shifts in the density decay curves (dashed lines) with increasing NaCl coverage with the experimentally determined apparent heights[6]. We obtain 4.1±0.4 Å for the first double layer and 2.2±0.4 Å for the next single layer, in reasonable agreement with the experimental values of 3.8±0.3 Å and 2.0±0.3 Å [133]. A closer inspection of the decay curves reveals an interesting explanation for the discrepancy between the geometrical height (i.e. the inter-layer separation, 2.8 Å) and the heights determined in STM (2.0 Å). Around the chlorine atoms, the density curves exhibit plateau-like regions which can be attributed to the hybridisation of the tails of the Ge states with the chlorine $3p$ states. Between the chloride layers, the density decays again exponentially with a constant similar to the vacuum (cf. Fig. D.6, top). The apparent height in STM is therefore mainly determined by the width of the hybridisation-induced plateaus, which is of course not directly linked to the chloride layer separation. This also implies that STM may not be able to detect vertical displacements of the ions within the film unless they affect the hybridisation width.

[6]Since the slope of the curves changes a little with film thickness (cf. Fig. D.6 top) the results depend on the density value chosen. The results are given for 10^{-10} e/bohr^3 and change by ∼0.2 Å per order order of magnitude.

Appendix E

Computer code developments

The *GW* space-time method as published in previous papers [73–75] has been implemented in the `gwst` code by Rex Godby and coworkers. When the Fortran code was made available to us, it became clear that

- the lack of program structure and its documentation would make extensions and modifications difficult,
- the code had been substantially modified with little or no documentations of the changes,
- major parts of the code had a suboptimal performance,
- the computation for large systems would be limited by the disk space requirements.[1]

In other words, the slab systems of this work were beyond the capabilities of this original version of the code. Here, we will explain how block algorithms were used to speed up the computation of the Green's function and the inversion of dielectric matrices, and how the disk space consumption could be reduced by 30-50%.

Before addressing the details, we note a few things about the technical background. On modern computer systems, algebraic operations are often more efficient when performed on blocks. The reason behind this is that the basic computational steps are no longer the bottleneck of the computation, but rather the data transfer between the main memory and the central processing unit (CPU). Block algorithms exploit the fact that small, but considerably faster memory chips (level caches) are used to buffer the data transfer.

[1]The storage requirements for the non-local operators of large systems exceeds the typical main memory sizes by an order of magnitude, taking into account that the code is not parallelised and is therefore run on work stations with ∼8GB main memory.

When all data for a calculation fits into this level cache, dramatic speedups are observed (often by a factor two or three), in particular for matrix-matrix operations. Unfortunately, most compilers are not able to transform standard high-level code ("plain Fortran") into optimally blocked machine code, and usually specialised libraries (basic linear algebra subroutines = BLAS) must be used. BLAS routines are classified into level 1 (vector operations), level 2 (matrix-vector), and level 3 (matrix-matrix). The speedup compared to conventional computer code increases with the BLAS level since the higher levels profit more from data reuse.

A second point concerns the memory management. Since the full size of the two-point functions exceeds by far the main memory of the work stations used for the calculation[2], only parts of it ("slices") can be kept in the main memory. Currently unused data is stored on the hard disk, which is typically 10–100 times larger than the main memory. A slice contains all points for one or more active indices for only one point of the inactive indices. Let's take a simple example with only two indices A and B with N_A and N_B many points, respectively. For transformations on A (e.g. a Fourier transformation), we use N_A-sized A-slices for one particular value of B. After the transformation has been performed, the slice is written into a "scratchfile" on the hard disk. One can then reuse the memory to perform the same transformation for the next B until the full $N_A \times N_B$ matrix has been transformed and written to the scratchfile. The data is then read back in a different order for the transformation of the previously inactive index B, i.e. we work with N_B-sized B-slices for one particular value of A. The limiting factor for the size of the two-point functions is then the available disk space. Moreover, the read/write operations take a significant amount of time and it is important to minimise these operations by carefully balancing the order of the transformations and the "layout", i.e. the index ordering, of the scratchfile.

E.1 Green's function

The construction of the Green's function is one of the time-critical steps in the *GW* space-time approach. In practice, the Green's function is computed in mixed space, i.e. $G_{\mathbf{k}}(\mathbf{r}, \mathbf{r}', i\tau)$. As explained above, this is done piecewise. In one step, G is computed for one specific $\mathbf{r}$ and $\mathbf{k}$-point, but all possible $\mathbf{r}'$

[2]The `gwst` code is a serial code, and therefore, parallel computers (shared memory model) or computer clusters (distributed memory model) which have much larger memories could not be used.

and τ, by summing over bands

$$G_{\mathbf{k}}(\mathbf{r}, \mathbf{r}', i\tau) = \sum_n \psi^*_{n\mathbf{k}}(\mathbf{r}) \psi_{n\mathbf{k}}(\mathbf{r}') f_{n\mathbf{k}}(\tau) \tag{E.1}$$

where ψ are the wavefunctions (which are already stored in a large block) and $f_{n\mathbf{k}}(\tau)$ are the precomputed frequency factors $\pm \exp(\pm \epsilon_{n\mathbf{k}} \tau)$. In the original "plain Fortran" code, this operation was performed in a triple loop over $\mathbf{r}'$, $\mathbf{n}$, and τ, where the performance bottleneck is the memory access.

However, the band summation can be reinterpreted as a matrix-matrix multiplication. By precomputing an auxiliary $N_{\text{bands}} \times N_\tau$ matrix

$$\eta(n, \tau) = \psi^*_{n\mathbf{k}}(\mathbf{r}) f_{n\mathbf{k}}(\tau) \ , \tag{E.2}$$

the summation over bands can be performed as a highly efficient matrix-matrix multiplication, schematically written as

$$G(N_{\mathbf{r}'} \times N_\tau) = \psi(N_{\mathbf{r}'} \times N_{\text{bands}}) \times \eta(N_{\text{bands}} \times N_\tau) \ , \tag{E.3}$$

which has been implemented as a BLAS level 3 call (`zgemm`). The resulting code is approximately two to three times faster than the original "plain Fortran" code.

For large N_{bands}, an additional trick allows to reduce the computational effort even more. The real-space representation of the wavefunctions $\psi_{n\mathbf{k}}(\mathbf{r})$ is computed from their Fourier representation

$$\psi_{n\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}} \sum_{\mathbf{G}} u_{n\mathbf{k}}(\mathbf{G}) e^{i\mathbf{G}\cdot\mathbf{r}} \tag{E.4}$$

using Fast Fourier Transforms (FFTs). Since a spherical cutoff is used in reciprocal space, the number of $\mathbf{G}$-vectors is considerably smaller than the number of real-space points (typically $N_{\mathbf{G}} \approx (0.3 - 0.5) N_{\mathbf{r}}$). As the Fourier transform for $\mathbf{r}'$ commutes with the band summation, it is advantageous to perform the sum over bands in reciprocal space for only $N_{\mathbf{G}}$ many points

$$G_{\mathbf{k}}(\mathbf{r}, \mathbf{G}', i\tau) = \sum_n \eta(n, \tau) u_{n\mathbf{k}}(\mathbf{G}') \ , \tag{E.5}$$

and perform the Fourier transformation from $\mathbf{G}'$ to $\mathbf{r}'$ afterwards

$$G_{\mathbf{k}}(\mathbf{r}, \mathbf{r}', i\tau) = e^{i\mathbf{k}\cdot\mathbf{r}'} \sum_{\mathbf{G}'} G_{\mathbf{k}}(\mathbf{r}, \mathbf{G}', i\tau) e^{i\mathbf{G}'\cdot\mathbf{r}'} \ . \tag{E.6}$$

Although this involves additional computational work for the Fourier transformation, the smaller matrix sizes for the expensive band summation reduce the overall computational cost significantly even for moderate N_{bands}. In practice, those $\psi_{n\mathbf{k}}(\mathbf{r})$ that are required for computing $\eta(n, \tau)$ are stored on disk and loaded on demand. This reciprocal-space summation is particularly helpful for high band cutoffs because in this case the GW calculation is dominated by the computation of the Green's function.

E.2 Block inversion of Hermitean packed matrices

The inversion of the dielectric matrices is another computationally demanding step in a *GW* calculation that we were able to speed up considerably by developing and implementing a block algorithm. The standard Hermitean packed-matrix inversion in the linear algebra package originally employed (LAPACK) uses an iterative scheme which achieves only 20-40% of the peak performance of the CPU. In the block algorithm developed here, the computation is reformulated in matrix-matrix operations that make full use of the performance of the underlying BLAS library.

We will demonstrate the block algorithm in the following for 'L'-packed matrices that store the lower triangle of the matrix column-wise. The corresponding algorithm for 'U'-packed matrices, where the upper triangle is stored, works analogously. Schematically, the blocked inversion (with a block size N_b) of a Hermitean $N \times N$ matrix can be written as

$$\begin{pmatrix} \mathbf{A} & \mathbf{B}^\dagger \\ \mathbf{B} & \mathbf{C} \end{pmatrix} \begin{pmatrix} \mathbf{a} & \mathbf{b}^\dagger \\ \mathbf{b} & \mathbf{c} \end{pmatrix} = \begin{pmatrix} \mathbf{1} & \mathbf{0}^\dagger \\ \mathbf{0} & \mathbf{1} \end{pmatrix} \tag{E.7}$$

where $\mathbf{A}$ is the $N_\mathrm{b} \times N_\mathrm{b}$ top left submatrix of the original matrix, $\mathbf{B}$ the $N_\mathrm{r} \times N_\mathrm{b}$ bottom left submatrix, and $\mathbf{C}$ the $N_\mathrm{r} \times N_\mathrm{r}$ bottom right submatrix. $N_\mathrm{r} = N - N_\mathrm{b}$ is the rest size of the matrix when the first N_b columns are separated. Small letters denote the corresponding submatrices of the inverse which are to be computed, and $\mathbf{1}$ and $\mathbf{0}$ on the right hand side are properly sized unit and zero matrices, respectively.

Equation (E.7) defines a set of four coupled matrix equations, which can be solved for the submatrices $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ of the inverse matrix:

$$\mathbf{c} = (\mathbf{C} - \mathbf{B}\mathbf{A}^{-\mathbf{1}}\mathbf{B}^\dagger)^{-\mathbf{1}} , \tag{E.8}$$

$$\mathbf{b} = -\mathbf{c}\mathbf{B}\mathbf{A}^{-\mathbf{1}} , \tag{E.9}$$

$$\mathbf{a} = \mathbf{A}^{-\mathbf{1}} - \mathbf{A}^{-\mathbf{1}}\mathbf{B}^\dagger\mathbf{b} . \tag{E.10}$$

The matrix inversion for the (small) matrix A is performed by a standard LAPACK routine. For the inversion in Eq. E.8, the blocked inversion routine is called recursively until the matrix is small enough for a standard LAPACK inversion. The size of the matrix is reduced by N_b for each level of recursion.

In practice, the result of the inversion is stored on the memory location of the original matrix. Moreover the intermediate steps should not require large amounts of additional memory. The scheme presented here requires only a $N \times N_\mathrm{b}$ work space, and uses also the input/output matrix memory

for intermediate results. The memory blocks, denoted by Greek letters, are listed in the following table:

name	size	storage	where
α	$N_\mathrm{b} \times N_\mathrm{b}$	Hermitean packed	input/output
β	$N_\mathrm{r} \times N_\mathrm{b}$	general	input/output
γ	$N_\mathrm{r} \times N_\mathrm{r}$	Hermitean packed	input/output
δ	$N_\mathrm{b} \times N_\mathrm{b}$	Hermitean unpacked	work space
ϵ	$N_\mathrm{r} \times N_\mathrm{b}$	general	work space

In the beginning, $\mathbf{A}$ and $\mathbf{B}$ are stored unseparated in α and β, indicated as $\mathbf{A}/\mathbf{B}$, and the same is true for $\mathbf{a}$ and $\mathbf{b}$ at the end.

The computational steps are listed in Table E.1. The mathematical operation to take is listed in the left column, the status of the various storage locations after the operation in the centre and the subroutine employed in the rightmost column. `hpgesub` either extracts (o) the bottom left submatrix from a Hermitean packed matrix, or stores it back (i). Likewise, `hphesub` performs the same operations for the top left square block (which is Hermitean). `hpinv` is the name of the blocked inversion routine that is called recursively to invert the $N_\mathrm{r} \times N_\mathrm{r}$ matrix γ. If the size of the matrix is smaller than N_b, the standard LAPACK iterative inversion (`hptrf` + `hptri`) is performed instead; and likewise for the inversion of the Hermitean unpacked matrix δ. `hemm` and `her2k` are standard BLAS matrix-matrix multiplication and rank-2 updates for Hermitean unpacked matrices. The corresponding subroutines for Hermitean packed matrices are missing from the standard BLAS and have been implemented using a block algorithm and available standard BLAS matrix-matrix routines (`hpmm_b` and `hpr2k_b`).

This blocked inversion algorithm has been implemented in the `gwst` code for the inversion of the dielectric matrix, one of the time-critical steps notably in large *GW* calculations. As expected from the performance of the underlying BLAS library, the blocked algorithm is faster by a factor 1.5–2.5 compared to the standard LAPACK routines because it uses matrix-matrix operations throughout.

The modifications described here (and others that are not described here) have greatly improved the computational efficiency of the `gwst` implementation. This is demonstrated in Fig. E.1. We observe a reduction in the overall run-time by a factor 3–5. The most important reduction of the computational time arises from the Green's function, the previously dominant part in the calculation. After the improvements, the computational effort is typically equally distributed between 1) the computation of the Green's function, 2) the Fourier transformation between the real-space/imaginary time and reciprocal-space/imaginary frequency representations, 3) the inversion

	result					
	α	β	γ	δ	ϵ	
	$\mathbf{A}/\mathbf{B}$		$\mathbf{C}$	–	–	
$\delta \leftarrow \mathbf{A}$	$\mathbf{A}/\mathbf{B}$		$\mathbf{C}$	$\mathbf{A}$	–	`hphesub(o)`
$\epsilon \leftarrow \mathbf{B}$	$\mathbf{A}/\mathbf{B}$		$\mathbf{C}$	$\mathbf{A}$	$\mathbf{B}$	`hpgesub(o)`
$\delta \leftarrow \delta^{-1}$	$\mathbf{A}/\mathbf{B}$		$\mathbf{C}$	$\mathbf{A^{-1}}$	$\mathbf{B}$	`hetrf+hetri`
$\alpha \leftarrow \delta$	$\mathbf{A^{-1}}$	–	$\mathbf{C}$	$\mathbf{A^{-1}}$	$\mathbf{B}$	`hphesub(i)`
$\beta \leftarrow \epsilon\delta$	$\mathbf{A^{-1}}$	$\mathbf{BA^{-1}}$	$\mathbf{C}$	$\mathbf{A^{-1}}$	$\mathbf{B}$	`hemm`
$\gamma \leftarrow \gamma - \epsilon\beta^\dagger$	$\mathbf{A^{-1}}$	$\mathbf{BA^{-1}}$	$\mathbf{C - BA^{-1}B^\dagger}$	–	$\mathbf{B}$	`hpr2k_b`[1]
$\gamma \leftarrow \gamma^{-1}$	$\mathbf{A^{-1}}$	$\mathbf{BA^{-1}}$	$\mathbf{c}$	–	–	`hpinv`[2] (Eq. E.8)
$\epsilon \leftarrow -\gamma\beta$	$\mathbf{A^{-1}}$	$\mathbf{BA^{-1}}$	$\mathbf{c}$	–	$\mathbf{b}$	`hpmm_b`[1] (Eq. E.9)
$\delta \leftarrow \alpha$	$\mathbf{A^{-1}}$	$\mathbf{BA^{-1}}$	$\mathbf{c}$	$\mathbf{A^{-1}}$	$\mathbf{b}$	`hphesub(o)`
$\delta \leftarrow \delta - \beta^\dagger\epsilon$	$\mathbf{A^{-1}}$	$\mathbf{BA^{-1}}$	$\mathbf{c}$	$\mathbf{a}$	$\mathbf{b}$	`her2k` (Eq. E.10)
$(\alpha\beta) \leftarrow \delta$	$\mathbf{a}/-$		$\mathbf{c}$	$\mathbf{a}$	$\mathbf{b}$	`hphesub(i)`
$(\alpha\beta) \leftarrow \epsilon$	$\mathbf{a}/\mathbf{b}$		$\mathbf{c}$	$\mathbf{a}$	$\mathbf{b}$	`hpgesub(i)`

[1] using δ as work space [2] using ϵ as work space

Table E.1: Computational steps and memory usage for the blocked inversion of Hermitean packed matrices. See text.

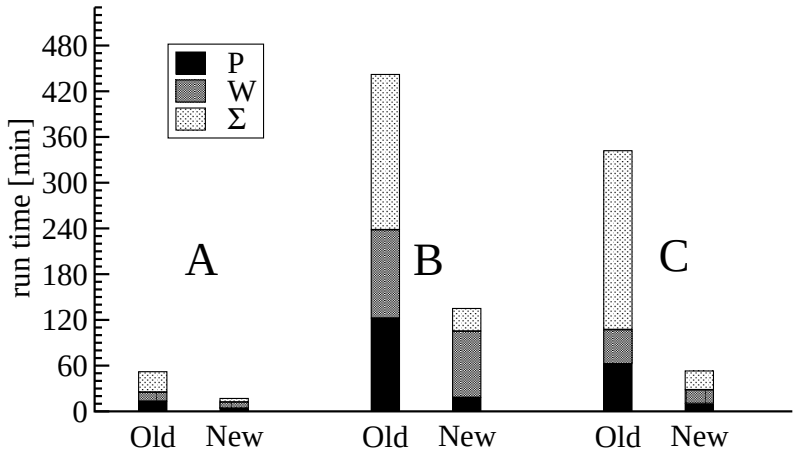

Figure E.1: Comparison of the original implementation of the space-time method and the improved version for three test cases: A) Si_4H_4 slab, 1 band-structure point. B) Na_2Cl_2 slab, 1 band structure point, band cutoff 5 Hartree. C) NaCl bulk, 8 band structure points, band cutoff 10 Hartree. The calculation can be separated into three distinct parts: computation of the polarisability in mixed space (P), computation of the screened interaction in mixed-space (W), computation of the self-energy and its matrix elements (Σ). In P and Σ, the Green's function is computed for which a typical speed-up by a factor of 6 could be achieved.

of the dielectric matrices, and 4) the computation of the self-energy matrix elements.

E.3 Disk storage

The size of the two-point functions in the *GW* formalism is in general too large to keep all the necessary data in the main memory. Most of the data is therefore kept on disk and only the currently required data is loaded into the main memory. After the optimisations described above (and many others not described here) the disk usage has proved to be the limiting factor for the calculations.

In the most disk-space efficient `grecomp`-mode[3] of the `gwst` program, an $(\mathbf{r}', \mathbf{R}, i\tau)$-slice of the Green's function for one special point $\mathbf{r}_s$ is computed from the wavefunctions whenever it is needed to compute the corresponding slice of the polarisability P or the self-energy Σ. The storage of the polarisability/the screened interaction is then the critical point. The key to reducing the disk space requirements is again the observation that the reciprocal space representation is 2–3 times smaller than the real-space representation.

The successive transformation of P from real space and imaginary time to reciprocal space and imaginary frequency is listed in Tab. E.2 together with the scratchfiles used. The backward transformation of W^{sr} is completely analogous. In the original implementation (highlighted in red), the main scratchfile 'cordata' contains the data after the first transformational step $\mathbf{R} \rightarrow \mathbf{k}^+$. We have introduced an alternative path (highlighted in blue) where the code proceeds until the Fourier transformation $\mathbf{r}' \rightarrow \mathbf{G}'$ before the data is stored on hard disk, thereby changing the content and the size of the scratchfile. Comparing the scratchfile size for the two alternatives listed in Tab. E.2, we find a common factor $N_{\mathbf{r}} N_\tau$, and differing factors $N_{\mathbf{r}_s} N_{\mathbf{k}^+}$ for the original layout and $N_{\mathbf{G}} N_{\mathbf{k}_s}$ for the new one. Which of the two is smaller is critically influenced by the symmetry reduction for $\mathbf{r}$ and $\mathbf{k}$, respectively, which depends on the relative number of high-symmetry points. For high symmetries, the symmetry reduction is usually more effective for $\mathbf{r}$ than for $\mathbf{k}$, which makes the original layout the optimal choice for highly symmetric bulk systems. However, when the number of symmetries is low as in our slab systems, the reciprocal-space reduction of $N_{\mathbf{G}}/N_{\mathbf{r}}$ favours the new layout and reduces the scratchfile size by 30–50% compared to the original layout. Only these reductions in disk space have made it possible to thoroughly check the convergence parameters beyond the values that were finally found to be

[3] `grecomp` stands for Green's function recomputed. The alternative is to store the Green's function on hard disk, too, thereby doubling the disk space requirements.

	representation		scratchfile	scratchfile size
	$\mathrm{P}(\mathbf{R},\mathbf{r}_s,\mathbf{r}',i\tau)$			
FFT	$\downarrow$			
	$\mathrm{P}(\mathbf{k}^+,\mathbf{r}_s,\mathbf{r}',i\tau)$	$\Longleftrightarrow$	cordata	$N_{\mathbf{k}^+} \times N_{\mathbf{r}_s} \times N_{\mathbf{r}} \times N_\tau$
rotate	$\downarrow$		$\downarrow$	
	$\mathrm{P}(\mathbf{k}_s,\mathbf{r}_s^*,\mathbf{r}',i\tau)$			
FFT	$\downarrow$			
	$\mathrm{F}(\mathbf{k}_s,\mathbf{r}_s^*,\mathbf{G}',i\tau)$	$\Longleftrightarrow$	cordata	$N_{\mathbf{k}_s} \times N_{\mathbf{r}} \times N_{\mathbf{G}} \times N_\tau$
reorder	$\downarrow$		$\uparrow$	
	$\mathrm{P}(\mathbf{k}_s,\mathbf{r},\mathbf{G}',i\tau)$	$\Longleftrightarrow^b$	semifft	$N_{\mathbf{r}} \times N_{\mathbf{G}}$
FFT	$\downarrow$			
	$\mathrm{P}(\mathbf{k}_s,\mathbf{G},\mathbf{G}',i\tau)$			
map	$\downarrow$			
	$\mathrm{P}(\mathbf{k}_s,[\mathbf{G}\mathbf{G}'],i\tau)$	$\Longleftrightarrow^a$	file_tw	$N_{\mathbf{G}\mathbf{G}'} \times \max(N_\tau, N_\omega)$
FT $\tau \rightarrow \omega$	$\downarrow$			
	$\mathrm{P}(\mathbf{k}_s,[\mathbf{G}\mathbf{G}'],i\omega)$	$\Longleftrightarrow^a$	file_tw	$N_{\mathbf{G}\mathbf{G}'} \times \max(N_\tau, N_\omega)$

[a] use of this scratchfile can be switched off ('twbio' option).
[b] improvement: not used when sufficient memory is available.

Indices:

$\mathbf{R}$	real space lattice vector
$\mathbf{k}^+$	Brillouin zone vector, reduced by time reversal symmetry
$\mathbf{k}_s$	special (symmetry-reduced) Brillouin zone vector
$\mathbf{r}$	real space vector (in unit cell)
$\mathbf{r}_s$	special (symmetry-reduced) vector (in unit cell)
$\mathbf{r}_s^*$	star of a special (symmetry-reduced) vector (in unit cell)
$\mathbf{G}$	reciprocal space lattice vector
$[\mathbf{G}\mathbf{G}']$	packed form of reciprocal space lattice vector matrix

Table E.2: Use of scratchfiles in the transformations from real space and imaginary time to reciprocal space and imaginary frequency. The red and blue colours refer to the old and new layout.

sufficient. In particular the plane-wave cutoff is a critical parameter, since the size of the two-point functions scales cubically with the cutoff.

E.4 Efficient computation of the matrix elements of r

Head and wings of the dielectric matrix at the Γ-point cannot be computed from the numerical polarisability matrix and the Coulomb potential directly. The small-**k** asymptotic behaviour of the polarisability (k^2 for the head, k for the wings) cancels with the corresponding singularity of the Coulomb potential and its square root, respectively. Therefore, these elements are obtained directly from the Kohn-Sham wavefunctions using a $\mathbf{k}\cdot\mathbf{p}$ perturbation approach [59, 86, 154]. For this purpose, the matrix elements of the position operator **r** are required, which are in practice calculated via the commutator of **r** with the Kohn–Sham Hamiltonian h^{KS}

$$\langle\psi_{c\mathbf{q}}|\mathbf{r}|\psi_{v\mathbf{q}}\rangle = \frac{\langle\psi_{c\mathbf{q}}|[h^{\mathrm{KS}},\mathbf{r}]|\psi_{v\mathbf{q}}\rangle}{\epsilon_{c\mathbf{q}}-\epsilon_{v\mathbf{q}}}\ . \tag{E.11}$$

The Kohn–Sham pseudopotential Hamiltonian consists of three parts: the effective potential, the kinetic energy and the non-local pseudopotential. The effective potential commutes with **r**. The contribution from the kinetic-energy operator $\frac{1}{2}\mathbf{p}^2$ is trivial to compute. Exploiting the commutator identity $[AB,C] = A[B,C] + [A,C]B$ with $A = B = \mathbf{p}$ and $C = \mathbf{r}$ yields

$$[\frac{1}{2}\mathbf{p}^2,\mathbf{r}] = \frac{1}{2}\left(\mathbf{p}[\mathbf{p},\mathbf{r}] + [\mathbf{p},\mathbf{r}]\mathbf{p}\right) = -i\mathbf{p}\ , \tag{E.12}$$

where we have made use of the fundamental commutator $[\mathbf{p},\mathbf{r}] = -i$. This is readily implemented for a plane-wave basis. The contribution from the non-local pseudopotential V_{nl} is more cumbersome and has often been neglected in earlier calculations. We will show in the following that it can be computed efficiently in a separable expression.

In its separable Kleinman–Bylander form [46], the non-local pseudopotential operator is written in the Dirac notation as

$$V_{\mathrm{nl}} = \sum_{\mu} |\chi_\mu\rangle E_\mu \langle\chi_\mu|\ , \tag{E.13}$$

where μ is a composed index $\{\mathbf{R}_\mu, n_\mu, l_\mu, m_\mu\}$ that runs over all pseudopotential projectors while χ_μ is in general given in a radial basis around a certain atomic position $\mathbf{R}_\mu$, i.e.,

$$\chi_\mu(\mathbf{r}) = f_{n_\mu l_\mu}(|\mathbf{r}-\mathbf{R}_\mu|)\ Y_{l_\mu m_\mu}(\Omega_{\mathbf{r}-\mathbf{R}_\mu})\ . \tag{E.14}$$

μ can additionally run over chemical species, which does not alter the following derivation, except that f_{nl} additionally depends on the species. We will now show that the matrix elements can be written in separable form, which reduces the scaling to be linear in the number of plane waves instead of quadratic as demonstrated in a previous approach [88].

To this end consider the commutator of $\mathbf{r}$ with a single projector:

$$\begin{aligned}
&\Big(|\chi_\mu\rangle E_\mu \langle\chi_\mu|\mathbf{r}\Big) \;-\; \Big(\mathbf{r}|\chi_\mu\rangle E_\mu\langle\chi_\mu|\Big) \\
&= E_\mu\,\Big[\Big(|\chi_\mu\rangle\langle\chi_\mu|\mathbf{r}\Big) \;-\; |\chi_\mu\rangle\mathbf{R}_\mu\langle\chi_\mu| \;+\; |\chi_\mu\rangle\mathbf{R}_\mu\langle\chi_\mu| \;-\; \Big(\mathbf{r}|\chi_\mu\rangle\langle\chi_\mu|\Big)\Big] \\
&= E_\mu\Big[|\chi_\mu\rangle\langle\chi_\mu|\,(\mathbf{r}-\mathbf{R}_\mu) \;-\; (\mathbf{r}-\mathbf{R}_\mu)\,|\chi_\mu\rangle\langle\chi_\mu|\Big] \qquad \text{(E.15)}
\end{aligned}$$

Now we make use of the fact that $\mathbf{r}-\mathbf{R}_\mu$ can be expressed in the same radial basis as χ_μ

$$[\mathbf{r}-\mathbf{R}_\mu]_\alpha = |\mathbf{r}-\mathbf{R}_\mu| \sum_{m=-1}^{1} c_{\alpha m} Y_{1m}(\Omega_{\mathbf{r}-\mathbf{R}_\mu})\;, \qquad \text{(E.16)}$$

where $\alpha \in \{x,y,z\}$ are the spatial directions, and $c_{\alpha m}$ yield the spatial components of the spherical harmonics for $l=1$:

$c_{\alpha m}$	$\alpha=x$	$\alpha=y$	$\alpha=z$
$m=-1$	$\frac{1}{\sqrt{2}}$	$\frac{i}{\sqrt{2}}$	0
$m=0$	0	0	1
$m=1$	$-\frac{1}{\sqrt{2}}$	$\frac{i}{\sqrt{2}}$	0

We can then write the product in the radial basis, too,

$$\begin{aligned}
|\chi^\alpha_\mu\rangle \;&:=\; [\mathbf{r}-\mathbf{R}_\mu]_\alpha|\chi_\mu\rangle \\
&= |\mathbf{r}-\mathbf{R}_\mu| \sum_{m=-1}^{1} c_{\alpha m}\; Y_{1m}(\Omega_{\mathbf{r}-\mathbf{R}_\mu}) f_{n_\mu l_\mu}(|\mathbf{r}-\mathbf{R}_\mu|)\; Y_{l_\mu m_\mu}(\Omega_{\mathbf{r}-\mathbf{R}_\mu}) \\
&= \sum_{L=l_\mu\pm1} \sum_{M=m_\mu-1}^{m_\mu+1} c^{L,l_\mu}_{\alpha M,m_\mu}\; f^r_{n_\mu l_\mu}(|\mathbf{r}-\mathbf{R}_\mu|)\; Y_{LM}(\Omega_{\mathbf{r}-\mathbf{R}_\mu}) \qquad \text{(E.17)}
\end{aligned}$$

with

$$\begin{aligned}
f^r_{nl}(\rho) \;&=\; \rho\; f_{nl}(\rho)\;, \qquad \text{(E.18)} \\
c^{L,l}_{\alpha M,m} \;&=\; \sum_{m'} c_{\alpha m'}(l\,m\;1\,m'|L\,M)\;. \qquad \text{(E.19)}
\end{aligned}$$

where $(l\,m\;1\,m'|L\,M)$ is a Clebsch–Gordan coefficient. It is convenient to expand χ^α_μ in a plane-wave basis similar to what is done for χ_μ. When the radial functions f_{nl} are given on a radial grid [43], f^r_{nl} is trivial to compute and

the same routines that are used to compute $\chi_\mu(\mathbf{G}+\mathbf{k})$ in the DFT calculation can be employed for the summands in $\chi^\alpha_\mu(\mathbf{G}+\mathbf{k})$. It must be emphasised that the sums over L and M contain only a very small number of non-zero terms (at most six).

The final formula is thus again a separable expression

$$[V_{\mathrm{nl}}, r_\alpha] = \sum_\mu E_\mu \left(|\chi_\mu\rangle\langle\chi^\alpha_\mu| - |\chi^\alpha_\mu\rangle\langle\chi_\mu| \right) . \tag{E.20}$$

The computational effort to set up a full $N_v \times N_c$ matrix for all three directions requires $(1+3)N_{\mathbf{G}}N_\mu(N_v+N_c)$ operations to calculate the $\langle\chi^\alpha_\mu|\psi_{v/c}\rangle$ projections and $6N_\mu N_v N_c$ operations to build up the 3 matrices from the projections in Equation (E.20). The scaling is thus linear in the number of $\mathbf{G}$-vectors $N_{\mathbf{G}}$ and not quadratic [88].

Only this efficient algorithm for the matrix elements of the position operator made the accurate computation of the dielectric tensor for the slab systems of interest feasible.

Appendix F

Computational parameters

F.1 Pseudopotentials

The pseudopotentials used in this work were generated with the `fhi98PP` program [43]. The parameters were varied from the default parameters in most cases to improve the accuracy and efficiency of the pseudopotentials. The optimised parameters are listed in the following table.

	occupation			r_c [a]				
	s	p	d	s	p	d	l_{loc}	remarks
Al	2	1	-	1.05(H)	(H)	1.4(H)	p	
Si	2	2	-	(H)	(H)	(H)	p	
Hf	2	6	2	0.75(H)	1.60(T)	0.9(H)	s	5-shell, 6s empty
O	1.7	4.8	-	1.35(T)	1.7(T)	(T)	d	
Ge	2	2	-	1.2(H)	(H)	(H)	p	
Na	1	0	-	2.7(T)	1.6(H)	2.5(T)	s	ρ_{cut} [b]=1.8
Cl	2	5	-	1.05(H)	1.1(H)	1.8(T)	p	
Mo	1	0	5	1.6(H)	(H)	2.44(T)	s	valence: 4d,5s,5p

[a] All radii in bohr. (T)=Troullier-Martins, (H)=Hamann. If no radius is specified, the fhi98PP default was used.

[b] density cut-off radius for non-linear core corrections.

F.2 DFT-LDA calculations

For DFT-LDA plane-wave calculations, there are two important convergence parameters: the plane-wave cutoff (that mainly depends on the pseudopotentials used) and the Monkhorst-Pack **k**-point folding which depends on the material investigated. These convergence parameters have been tested for each system and the parameters employed for the SCF calculations are listed

in the following table:

system	E_{cut} [Ry]	**k**-point grid
SiO_2 bulk	60	$3 \times 3 \times 3$
SiO_2 slabs	60	$3 \times 3 \times 1$
Mo bulk	60	$22 \times 22 \times 22$
SiO_2/Mo	60	$6 \times 6 \times 1$
Al_2O_3 bulk	60	$3 \times 3 \times 3$
Al_2O_3 slabs	60	$3 \times 3 \times 1$
HfO_2 bulk	60	$4 \times 4 \times 4$
HfO_2 slabs	60	$4 \times 4 \times 1$
NaCl bulk	40	$3 \times 3 \times 3$
NaCl slabs	40	$4 \times 4 \times 1$
Ge bulk	20	$4 \times 4 \times 4$
NaCl/Ge	40	$6 \times 3 \times 1$

F.3 G_0W_0 calculations

F.3.1 Time-frequency grids

The time-frequency grids are characterised by two parameters: the maximum time or frequency for the numerical part and the number of Gauss-Legendre points used per half-axis [75]. We have tested the grids for each of the bulk systems investigated in this work, i.e. Ge, SiO_2, NaCl, Al_2O_3, and HfO_2. In all cases, a maximum of 6 atomic units and 15 Gauss-Legendre points proved to be sufficient to achieve an accuracy below 0.05 eV.

F.3.2 System-dependent convergence parameters

The following table summarises further convergence parameters for the G_0W_0 calculations. They were tested and give results within 0.05 eV for each parameter, except for NaCl/Ge where the achieved accuracy is 0.05–0.1 eV.

We will briefly comment on the various parameters. The plane-wave (pw) cutoff determines the real-space resolution of the two-point functions and is the most critical parameter for the computational effort and the required memory and disk space. The G_0W_0 calculation in the space-time method scales cubically with the pw cutoff (in reciprocal space, the computational effort in some parts even scales with a power 4.5). The quasiparticle energies usually vary non-monotoneously with the pw cutoff, the absolute accuracy can therefore only be estimated. Usually, the bare exchange part of the self-energy requires a higher pw cutoff than the correlation part, which can be

parameter	NaCl bulk	NaCl slabs	NaCl/Ge
pw cutoff Σ_x [Hartree]	14	14	10
pw cutoff Σ_c [Hartree]	14	10	7
k-sampling	$4 \times 4 \times 4$	–	6×3
k-sampling head/wings	$4 \times 4 \times 4$	4×4	10×10
bcut head/wings [Hartree]	2	2	1

Table F.1: Convergence parameters for the *GW* calculations (see text).

used to drastically reduce the computational effort when the two parts are determined independently.

Head and wings of the dielectric matrix at the Γ-point are computed via perturbation theory directly from the Kohn-Sham wavefunctions [86]. Since the convergence behaviour for the band cutoff and the **k**-point sampling differs between head, wings, and body of the dielectric matrix [86, 154], we determine the parameters for head and wings independently. Typically, we converge the dielectric tensor at the smallest frequency to 0.3–1% and assume that this is sufficient for the wings, too. Earlier test calculations in our group indicate that variations in the dielectric tensor of up to 10% introduce errors below 0.1 eV for semiconductor systems [56]. We note that head and wings include the contributions of the non-local pseudopotentials [86] for which we have developed a highly efficient algorithm, see Sec. E.4. This is essential to obtain a consistent dielectric tensor for the anisotropy treatment, cf. Sec. 3.2.2.

The importance of the **k**-point sampling in slab systems has been discussed in Sec. 3.3.2 and been explicitly studied for the free-standing NaCl films. For NaCl/Ge, the extrapolation technique was used only for the 2ML case to determine the correction term given in Sec. 5.2.3, otherwise the $6 \times 3 \times 1$ sampling as listed in the table was used.

F.3.3 Band cutoff

The convergence of NaCl with respect to the band cutoff *bcut* is shown in Fig. F.1 for the bulk and a slab, respectively. As for most systems, the convergence shows a $1/bcut$ behaviour and the data is plotted correspondingly. Fig. F.1 illustrates that NaCl requires a band cutoff of about 10 Hartree for a convergence to within 0.1 eV of the extrapolated value. This is much higher than for semiconductors, where 2–4 Hartree prove to be sufficient for this accuracy, see e.g. [23, 56, 74]. However, an absolute convergence is usually not required. Using a simple linear extrapolation yields highly reliable results in comparison with extended convergence tests for bulk systems. Nevertheless,

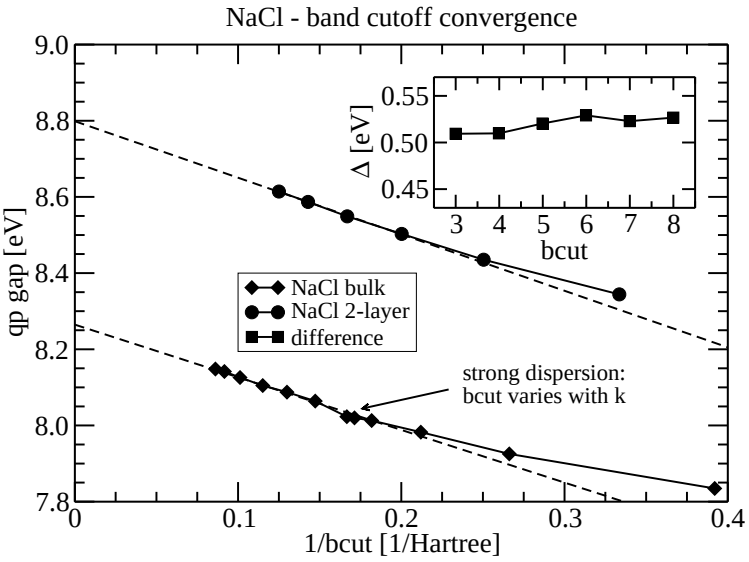

Figure F.1: Convergence of the quasiparticle gap for NaCl bulk and a 2-layer slab. The inset show the difference Δ between the two curves and demonstrates that it is practically independent of *bcut*.

the data in Fig. F.1 shows also that a careful inspection of the results is necessary in some cases. In practice, the band summations for the Green's function are not truncated at a certain energy, but at a certain band index. The highest band energy of all **k**-points then defines the nominal band cutoff which was used in the plot. In NaCl, there are highly dispersive bands around 5-6 Hartree which vary in energy by ~ 0.5 Hartree. Correspondingly, the effective band cutoff depends strongly on the **k**-point and differs considerably from the nominal band cutoff. It must be emphasized that this is a rather rare situation typical for small high-symmetry systems and does not affect the converged result.

Furthermore, we note that the underconvergence of the quasiparticle gap appears to be largely independent of the long-range order. It can be clearly seen that convergence curves for the bulk and slab are essentially parallel. This becomes obvious when considering the difference between the two curves, shown in the inset in Fig. F.1. The variations are below 0.025 eV and result mainly from the variations in the bulk. We can therefore extract the underconvergence with respect to the band cutoff from the bulk and transfer it to the slab systems, thus reducing the computational effort drastically. For the free-standing slab systems, a band cutoff of 5 Hartree was used and the correction to obtain the extrapolated value amounts to 0.28 eV.

Since the bare exchange self-energy and the DFT exchange-correlation

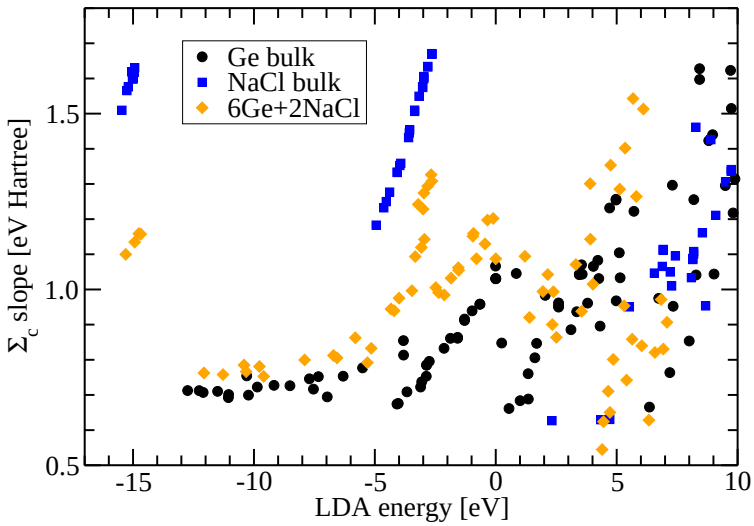

Figure F.2: Comparison of the Σ_c slope parameter for three systems in comparison: bulk NaCl, bulk Ge, and a Ge-supported NaCl film system.

potential are independent of the band summation, the variations exclusively affect the matrix elements of the correlation self-energy. For the supported films, we directly assume a $1/bcut$ behaviour

$$\Sigma_c(\text{bcut}) = \Sigma_c(\infty) + \frac{A}{\text{bcut}} \,. \tag{F.1}$$

Using a few cutoff energies, we can then determine the slope parameter A for each band from a linear regression. We note that the *bcut*-dependence of the gap results from the difference of the slope parameter for the band edge states, but does not depend on the absolute value. Indeed, comparing the slope parameters between different systems reveals that the low band cutoffs in the semiconductors must be attributed to the similarity of the slope parameters for the different bands (cf. F.2). The absolute values have a similar magnitude as for the ionic systems, about 0.5–2 eV·Hartree. For the calculations presented in Sec. 5.2.3, we then used a band cutoff of 3 Hartree.