6.4 Pseudo-potentials

The electrons in inner, fully occupied shells do not contribute in the chemical bindings. To simplify the calculations they can be replaced by pseudo-potentials, modeling the inner shells. Only the outer shells (including the valence shells) are then modeled using basis functions. The pseudo-potentials are chosen such that calculations for isolated atoms are as accurate as possible.

6.5 Effective models

To understand the properties of these materials the Hamilton operator of the full quantum chemical problem (6.1) is usually simplified to effective models, which still contain the same important features, but which are easier to investigate. They can be used to understand the physics of these materials, but not directly to quantitatively fit experimental measurements.

6.5.1 The tight-binding model

The simplest model is the tight-binding model, which concentrates on the valence bands. All matrix elements t_{ij} in equation (6.3), apart from the ones between nearest neighbor atoms are set to zero. The others are simplified, as in:

$$H = \sum_{\langle i,j\rangle,\sigma} (t_{ij} c^\dagger_{i,\sigma} c_{j,\sigma} + \text{H.c.}). \tag{6.17}$$

This model is easily solvable by Fourier transforming it, as there are no interactions.

6.5.2 The Hubbard model

To include effects of electron correlations, the Hubbard model includes only the often dominant intra-orbital repulsion V_{iiii} of the V_{ijkl} in equation (6.4):

$$H = \sum_{\langle i,j\rangle,\sigma} (t_{ij} c^\dagger_{i,\sigma} c_{j,\sigma} + \text{H.c.}) + \sum_i U_i n_{i,\uparrow} n_{i,\downarrow}. \tag{6.18}$$

The Hubbard model is a long-studied, but except for the 1D case still not completely understood model for correlated electron systems.

In contrast to band insulators, which are insulators because all bands are either completely filled or empty, the Hubbard model at large U is insulating at half filling, when there is one electron per orbital. The reason is the strong Coulomb repulsion U between the electrons, which prohibit any electron movement in the half filled case at low temperatures.

6.5.3 The Heisenberg model

In this insulating state the Hubbard model can be simplified to a *quantum* Heisenberg model, containing exactly one spin per site.

$$H = \sum_{\langle i,j \rangle} J_{ij} \vec{S}_i \vec{S}_j \tag{6.19}$$

For large U/t the perturbation expansion gives $J_{ij} = 2t_{ij}^2(1/U_i + 1/U_j)$. The Heisenberg model is the relevant effective models at temperatures $T \ll t_{ij}, U$ (10^4 K in copper oxides). The derivation will be shown in the lecture.

6.5.4 The t-J model

The t-J model is the effective model for large U at low temperatures away from half-filling. Its Hamiltonian is

$$H = \sum_{\langle i,j \rangle, \sigma} \left[(1 - n_{i,-\sigma}) t_{ij} c_{i,\sigma}^\dagger c_{j,\sigma} (1 - n_{j,-\sigma}) + \text{H.c.} \right] + \sum_{\langle i,j \rangle} J_{ij} (\vec{S}_i \vec{S}_j - n_i n_j / 4). \tag{6.20}$$

As double-occupancy is prohibited in the t-J model there are only three instead of four states per orbital, greatly reducing the Hilbert space size.

6.6 Exact diagonalization

The most accurate method is exact diagonalization of the Hamiltonian matrix using the Lanczos algorithm, discussed in appendix A.2. The size of the Hilbert space of an N-site system (4^N for a Hubbard model , 3^N for a t-J model and $(2S+1)^N$ for a spin-S model) can be reduced by making use of symmetries. Translational symmetries can be employed by using Bloch waves with fixed momentum as basis states. Conservation of particle number and spin allows to restrict a calculation to subspaces of fixed particle number and magnetization.

As an example we will sketch how to implement exact diagonalization for a simple one-dimensional spinless fermion model with nearest neighbor hopping t and nearest neighbor repulsion V:

$$H = -t \sum_{i=1}^{L-1} (c_i^\dagger c_{i+1} + \text{H.c.}) + V \sum_{i=1}^{L-1} n_i n_{i+1}. \tag{6.21}$$

The first step is to construct a basis set. We describe a basis state using "multi-bit coding". A many-body state of fermions can be represented as an unsigned integer where bit i set to one corresponds to an occupied site i. For spinful fermions we take either two integers, one for the up and one for the down spins, or two bits per site.

As the Hamiltonian conserves the total particle number we thus want to construct a basis of all states with N particles on L sites (or N bits set to one in L bits). In the code fragment below we use the following variables:

- `states_` is a vector storing the integers whose bit patterns correspond to the basis states. It can be accessed using the following functions:
  - `dimension()` returns the number of basis states.
  - `state(i)` returns the i-th basis state, where i runs from 0 to `dimension()`-1.
- `index_` is a much larger vector of size 2^L. It is used to obtain the number of a state in the basis, given the integer representation of the state. It can be accessed using the function
  - `index(s)` which returns the index i of the state in the basis, or the largest integer to denote an invalid state, if the bit pattern of the integer does not correspond to a basis state.

  Since this vector is very large, it will limit the size of system that can be studied. To save space, the `index_` array could be omitted and the `index(s)` function implemented by a binary search on the `states_` array.

Here is the C++ code for this class:

```
#include <vector>
#include <alps/bitops.h>
#include <limits>
#include <valarray>
#include <cassert>

class FermionBasis {
public:
  typedef unsigned int state_type;
  typedef unsigned int index_type;
  FermionBasis (int L, int N);

  state_type state(index_type i) const {return states_[i];}
  index_type index(state_type s) const {return index_[s];}
  unsigned int dimension() const { return states_.size();}

private:
  std::vector<state_type> states_;
  std::vector<index_type> index_;
};
```

In the constructor we build the basis states. For N spinless fermions on L sites the valid basis states are all the ways to place N particles on L sites, which is equivalent to all integers between 0 and $2^L - 1$ that have N bits set. The constructor uses the `alps::popcnt` function of the ALPS library.

```
FermionBasis::FermionBasis(int L, int N)
{
  index_.resize(1<<L); // 2^L entries
  for (state_type s=0;s<index_.size();++s)
    if(alps::popcnt(s)==N) {
      // correct number of particles
          states_.push_back(s);
          index_[s]=states_.size()-1;
    }
    else
      // invalid state
      index_[s]=std::numeric_limits<index_type>::max();
}
```

Finally we want to implement a matrix-vector multiplication $v = Hw$ for our Hamiltonian and derive a Hamiltonian class. We do not want to store the matrix at all, neither in dense nor in sparse form but instead implement a fast function to perform the matrix-vector multiplication on-the-fly.

```
class HamiltonianMatrix : public FermionBasis {
public:
  HamiltonianMatrix(int L, int N, double t, double V)
    : FermionBasis(L,N), t_(t), V_(V), L_(L) {}

  void multiply(std::valarray<double>& v, const std::valarray<double>& w);

private:
  double t_, V_;
  int L_;
};
```

Finally we show the implementation of the matrix-vector multiplication. It might look like magic but we will explain it all in detail during the lecture.

```
void HamiltonianMatrix::multiply(std::valarray<double>& v,
               const std::valarray<double>& w)
{
  // check dimensions
  assert(v.size()==dimension());
  assert(w.size()==dimension());

  // do the V-term
  for (int i=0;i<dimension();++i) {
    state_type s = state(i);
    // count number of neighboring fermion pairs
    v[i]=w[i]*V_*alps::popcnt(s&(s>>1));
  }

  // do the t-term
  for (int i=0;i<dimension();++i) {
    state_type s = state(i);
    // inside the chain
    for (int r=0;r<L_-1;++r) {
      state_type shop = s^(3<<r);   // exchange two neighbors
      index_type idx = index(shop); // get the index
      if(idx!=std::numeric_limits<index_type>::max())
        v[idx]+=-t_*w[i];
    }
    // across the boundary
     state_type shop = s^(1|(1<<(L-1))); // exchange the first and last
     index_type idx = index(shop);       // get the index
      if(idx!=std::numeric_limits<index_type>::max())
        // watch out for Fermi sign since we hop over some particles
        v[idx]+=-t*(alps::popcnt(s&((1<<(L-1))-1))%2==0 ? 1 : -1)*w[i];
  }
}
```

This class can now be used with the Lanczos algorithm to calculate the energies and wave functions of the low lying states of the Hamiltonian.

In production codes one uses all symmetries to reduce the dimension of the Hilbert space as much as possible. In this example translational symmetry can be used if periodic boundary conditions are applied. The implementation gets much harder then.

In order to make the implementation of exact diagonalization much easier we have generalized the expression templates technique developed by Todd Veldhuizen for array expression to expressions including quantum operators. Using this expression template library we can write a multiplication

$$|\psi\rangle = H|\phi\rangle = (-t\sum_{i=1}^{L-1}(c_i^\dagger c_{i+1} + \text{H.c.}) + V\sum_{i}^{L-1} n_i n_{i+1})|\phi\rangle \tag{6.22}$$

simply as:

```
Range i(1,L-1);
psi = sum(i,(-t*(cdag(i)*c(i+1)+HermitianConjugate)+V*n(i)*n(i+1))*phi);
```

The advantage of the above on-the-fly calculation of the matrix in the multiplication routine is that the matrix need not be stored in memory, which is an advantage for the biggest systems where just a few vectors of the Hilbert space will fit into memory.

If one is not as demanding and wants to simulate a slightly smaller system, where the (sparse) matrix can be stored in memory, then a less efficient but more flexible function can be used to create the matrix and store it in memory. Such a program is available through the ALPS project at `http://alps.comp-phys.org/`. It allows to perform the above calculation just by describing the lattice and model in an XML input file.

6.7 The Hartree Fock method

6.7.1 The Hartree-Fock approximation

The Hartree-Fock approximation is based on the assumption of independent electrons. It starts from an ansatz for the N-particle wave function as a Slater determinant of N single-particle wave functions:

$$\Phi(\vec{r}_1, \sigma_1; \ldots; \vec{r}_N, \sigma_N) = \frac{1}{\sqrt{N}} \begin{vmatrix} \phi_1(\vec{r}_1, \sigma_1) & \cdots & \phi_N(\vec{r}_1, \sigma_1) \\ \vdots & & \vdots \\ \phi_1(\vec{r}_N, \sigma_N) & \cdots & \phi_N(\vec{r}_N, \sigma_N) \end{vmatrix} . \tag{6.23}$$

The orthogonal single particle wave functions ϕ_μ are chosen so that the energy is minimized.

For numerical calculations a finite basis has to be introduced, as discussed in the previous section. Quantum chemists distinguish between the self-consistent-field (SCF) approximation in a finite basis set and the Hartree-Fock (HF) limit, working in a complete basis. In physics both are known as Hartree-Fock approximation.

6.7.2 The Hartree-Fock equations in nonorthogonal basis sets

It will be easiest to perform the derivation of the Hartree-Fock equations in a second quantized notation. To simplify the discussion we assume closed-shell conditions, where each orbital is occupied by both an electron with spin $\uparrow$ and one with spin $\downarrow$. We start by writing the Hartree Fock wave function (6.23) in second quantized form:

$$|\Phi\rangle = \prod_{\mu,\sigma} c^\dagger_{\mu\sigma} |0\rangle, \tag{6.24}$$

where $c^\dagger_{\mu\sigma}$ creates an electron in the orbital $\phi_\mu(\mathbf{r}, \sigma)$. As these wave functions are orthogonal the $c^\dagger_{\mu\sigma}$ satsify the usual fermion anticommutation relations. Greek subscripts refer to the Hartree-Fock single particle orbitals and roman subscripts to the single particle basis functions. Next we expand the $c^\dagger_{\mu\sigma}$ in terms of the creation operators $\hat{a}^\dagger_{n\sigma}$ of our finite basis set:

$$c^\dagger_{\mu\sigma} = \sum_{n=1}^{L} d_{\mu n} \hat{a}^\dagger_{n\sigma} \tag{6.25}$$

and find that

$$a_{j\sigma}|\Phi\rangle = a_{j\sigma} \prod_{\mu,\sigma'} c^\dagger_{\mu\sigma'} |0\rangle = \sum_\nu d_{\nu j} \prod_{\mu\sigma' \neq \nu\sigma} c^\dagger_{\mu\sigma'} |0\rangle. \tag{6.26}$$

In order to evaluate the matrix elements $\langle\Phi|H|\Phi\rangle$ of the Hamiltonian (6.5) we introduce the bond-order matrix

$$P_{ij} = \sum_\sigma \langle\Phi| a^\dagger_{i\sigma} a_{j\sigma} |\Phi\rangle = 2 \sum_\nu d^*_{\nu i} d_{\nu j}, \tag{6.27}$$

where we have made use of the closed-shell conditions to sum over the spin degrees of freedom. The kinetic term of H is now simply $\sum_{ij} P_{ij} t_{ij}$. Next we rewrite the interaction part $\langle\Phi|a^\dagger_{i\sigma}a^\dagger_{k\sigma'}a_{l\sigma'}a_{j\sigma}|\Phi\rangle$ in terms of the P_{ij}. We find that if $\sigma=\sigma'$

$$\langle\Phi|a^\dagger_{i\sigma}a^\dagger_{k\sigma}a_{l\sigma}a_{j\sigma}|\Phi\rangle = \langle\Phi|a^\dagger_{i\sigma}a_{j\sigma}|\Phi\rangle\langle\Phi|a^\dagger_{k\sigma}a_{l\sigma}|\Phi\rangle - \langle\Phi|a^\dagger_{i\sigma}a_{l\sigma}|\Phi\rangle\langle\Phi|a^\dagger_{k\sigma}a_{j\sigma}|\Phi\rangle \tag{6.28}$$

and if $\sigma\neq\sigma'$:

$$\langle\Phi|a^\dagger_{i\sigma}a^\dagger_{k\sigma'}a_{l\sigma'}a_{j\sigma}|\Phi\rangle = \langle\Phi|a^\dagger_{i\sigma}a_{j\sigma}|\Phi\rangle\langle\Phi|a^\dagger_{k\sigma'}a_{l\sigma'}|\Phi\rangle \tag{6.29}$$

Then the energy is (again summing over the spin degrees of freedom):

$$E_0 = \sum_{ij} t_{ij}P_{ij} + \frac{1}{2}\sum_{ijkl}\left(V_{ijkl}-\frac{1}{2}V_{ilkj}\right)P_{ij}P_{kl}. \tag{6.30}$$

We now need to minimize the energy E_0 under the condition that the $|\phi_\mu\rangle$ are normalized:

$$1 = \langle\phi_\mu|\phi_\mu\rangle = \sum_{i,j} d^*_{\mu i}d_{\mu j}S_{ij}. \tag{6.31}$$

Using Lagrange multipliers to enforce this constraint we have to minimize

$$\sum_{ij} t_{ij}P_{ij} + \frac{1}{2}\sum_{ijkl}\left(V_{ijkl}-\frac{1}{2}V_{ilkj}\right)P_{ij}P_{kl} - \sum_\mu \epsilon_\mu \sum_{i,j} d^*_{\mu i}d_{\mu j}S_{ij} \tag{6.32}$$

Setting the derivative with respect to $d_{\mu i}$ to zero we end up with the Hartree-Fock equations for a finite basis set:

$$\sum_{j=1}^{L}(f_{ij}-\epsilon_\mu S_{ij})d_{\mu j} = 0, \tag{6.33}$$

where

$$f_{ij} = t_{ij} + \sum_{kl}\left(V_{ijkl}-\frac{1}{2}V_{ilkj}\right)P_{kl}. \tag{6.34}$$

This is again a generalized eigenvalue problem of the form $Ax=\lambda Bx$ and looks like a one-particle Schrödinger equation. However, since the potential depends on the solution it is a nonlinear and not a linear eigenvalue problem. The equation is solved iteratively, always using the new solution for the potential, until convergence to a fixed point is achieved.

The eigenvalues ϵ_μ of f do not directly correspond to energies of the orbitals, as the Fock operator counts the V-terms twice. Thus we obtain the total ground state energy from the Fock operator eigenvalues by subtracting the double counted part:

$$E_0 = \sum_{\mu=1}^{N}\epsilon_\mu - \frac{1}{2}\sum_{ijkl}\left(V_{ijkl}-\frac{1}{2}V_{ilkj}\right)P_{ij}P_{kl} \tag{6.35}$$

6.7.3 Configuration-Interaction

The approximations used in Hartree-Fock and density functional methods are based on non-interacting electron pictures. They do not treat correlations and interactions between electrons correctly. To improve these methods, and to allow the calculation of excited states, often the "configuration-interaction" (CI) method is used.

Starting from the Hartree-Fock ground state

$$|\psi_{HF}\rangle = \prod_{\mu=1}^{N} c_\mu^\dagger |0\rangle \tag{6.36}$$

one or two of the $c_\mu^\dagger$ are replaced by other orbitals $c_i^\dagger$:

$$|\psi_0\rangle = \left(1 + \sum_{i,\mu} \alpha_\mu^i c_i \dagger c_\mu + \sum_{i<j,\mu<\nu} \alpha_{\mu\nu}^{ij} c_i \dagger c_j \dagger c_\mu c_\nu \right) |\psi_{HF}\rangle. \tag{6.37}$$

The energies are then minimized using this variational ansatz. In a problem with N occupied and M empty orbitals this leads to a matrix eigenvalue problem with dimension $1 + NM + N^2M^2$. Using the Lanczos algorithm the low lying eigenstates can then be calculated in $\mathrm{O}((N+M)^2)$ steps.

Further improvements are possible by allowing more than only double-substitutions. The optimal method treats the full quantum problem of dimension $(N+M)!/N!M!$. Quantum chemists call this method "full-CI". Physicists simplify the Hamilton operator slightly to obtain simpler models with fewer matrix elements, and call that method "exact diagonalization". This method will be discussed later in the course.

6.8 Density functional theory

Another commonly used method, for which the Nobel prize in chemistry was awarded to Walter Kohn, is the density functional theory. In density functional theory the many-body wave function living in $\mathbb{R}^{3N}$ is replaced by the electron density, which lives just in $\mathbb{R}^3$. Density functional theory again reduces the many body problem to a one-dimensional problem. In contrast to Hartree-Fock theory it has the advantage that it could – in principle – be exact if there were not the small problem of the unknown exchange-correlation functional.

It is based on two fundamental theorems by Hohenberg and Kohn. The first theorem states that the ground state energy E_0 of an electronic system in an external potential V is a functional of the electron density $\rho(\vec{r})$:

$$E_0 = E[\rho] = \int d^3\vec{r} V(\vec{r})\rho(\vec{r}) + F[\rho], \tag{6.38}$$

with a universal functional F. The second theorem states that the density of the ground state wave function minimizes this functional. The proof of both theorems will be shown in the lecture.

These theorems make our life very easy: we only have to minimize the energy functional and we obtain both the ground state energy and the electron density in the ground state – and everything is exact!

The problem is that, while the functional F is universal, it is also unknown! Thus we need to find good approximations for the functional. One usually starts from the ansatz:

$$F[\rho] = E_h[\rho] + E_k[\rho] + E_{xc}[\rho]. \tag{6.39}$$

The Hartree-term E_h given by the Coulomb repulsion between two electrons:

$$E_h[\rho] = \frac{e^2}{2} \int d^3\vec{r} d^3\vec{r}' \frac{\rho(\vec{r})\rho(\vec{r}')}{|\vec{r} - \vec{r}'|}. \tag{6.40}$$

The kinetic energy $E_k[\rho]$ is that of a non-interacting electron gas with the same density. The exchange- and correlation term $E_{xc}[\rho]$ contains the remaining unknown contribution, which we will discuss a bit later.

To calculate the ground state density we have to minimize this energy, solving the variational problem

$$0 = \delta E[\rho] = \int d^3\vec{r}\delta\rho(\vec{r}) \left(V(\vec{r}) + e^2 \int d^3\vec{r}' \frac{\rho(\vec{r}')}{|\vec{r} - \vec{r}'|} + \frac{\delta E_k[\rho]}{\delta\rho(\vec{r})} + \frac{\delta E_{xc}[\rho]}{\delta\rho(\vec{r})} \right) \tag{6.41}$$

0 subject to the constraint that the total electron number to be conserved

$$\int d^3\vec{r}\delta\rho(\vec{r}) = 0. \tag{6.42}$$

Comparing this variational equation to the one for noninteracting system

$$\left(-\frac{1}{2m}\nabla^2 + V_{eff}(\vec{r}) \right) \phi_\mu(\vec{r}) = \epsilon_\mu \phi_\mu(\vec{r}), \tag{6.43}$$

we realize that they are the same if we define the potential of the non-interacting system as

$$V_{eff}(\vec{r}) = V(\vec{r}) + e^2 \int d^3\vec{r}' \frac{\rho(\vec{r}')}{|\vec{r} - \vec{r}'|} + v_{xc}(\vec{r}), \tag{6.44}$$

where the exchange-correlation potential is defined by

$$v_{xc}(\vec{r}) = \frac{\delta E_{xc}[\rho]}{\delta\rho(\vec{r})}. \tag{6.45}$$

The form (6.43) arises because we have separated the kinetic energy of the non-interacting electron system from the functional. The variation of this kinetic energy just gives the kinetic term of this Schrödinger-like equation.

The non-linear equation is again solved iteratively, making an ansatz using $N/2$ normalized single-electron wave functions, which we occupy with spin $\uparrow$ and spin $\downarrow$ electrons to get the electron density.

$$\rho(\vec{r}) = 2 \sum_{\mu=1}^{N/2} |\phi_\mu(\vec{r})|^2, \tag{6.46}$$

6.8.1 Local Density Approximation

Apart from the restricted basis set everything was exact up to this point. As the functional $E_{xc}[\rho]$ and thus the potential $v_{xc}(\vec{r})$ is not known, we need to introduce approximations.

The simplest approximation is the "local density approximation" (LDA), which replaces v_{xc} by that of a uniform electron gas with the same density. Instead of taking a functional $E[\rho](\vec{r})$ which could be a function of $\rho(\vec{r}), \nabla\rho(\vec{r}), \nabla\nabla\rho(\vec{r}), \dots$ we ignore all the gradients and just take the local density

$$E_{xc}[\rho](r) = E_{\text{LDA}}(\rho(r)); \tag{6.47}$$

Defining

$$r_s^{-1} = a_B \left(\frac{4\pi}{3}\rho\right)^{1/3} \tag{6.48}$$

the exchange correlation potential is

$$v_{xc} = -\frac{e^2}{a_B}\left(\frac{3}{2\pi}\right)^{2/3}\frac{1}{r_s}\left[1 + 0.0545 r_s \ln(1 + 11.4/r_s)\right] \tag{6.49}$$

where the first part corresponds to uncorrelated electrons and the last factor is a correlation correction determined by fitting to quantum Monte Carlo (QMC) simulations of an electron gas.

6.8.2 Improved approximations

Improvements over the LDA have been an intense field of research in quantum chemistry. I will just mention two improvements. The "local spin density approximation" (LSDA) uses separate densities for electrons with spin $\uparrow$ and $\downarrow$. The "generalized gradient approximation" (GGA) and its variants use functionals depending not only on the density, but also on its derivatives.

6.9 Car-Parinello molecular dynamics

In the lecture on "Computational Statistical Physics" you have learned about the molecular dynamics method, in which atoms move on classical trajectories under forces, such as those from the Lennard-Jones potential, which have been previously calculated in quantum mechanical simulations. It would be nicer, and more accurate, to use a full quantum mechanical force calculation at every time step instead of using such static forces that have been extracted from previous simulations.

Roberto Car (currently in Princeton) and Michele Parinello (currently at ETH) have combined density functional theory with molecular dynamics to do just that. Their method, Car-Parinello molecular dynamics (CPMD) allows much better simulations of molecular vibration spectra and of chemical reactions.

The atomic nuclei are propagated using classical molecular dynamics, but the electronic forces which move them are estimated using density functional theory:

$$M_n \frac{d^2\vec{R}_n}{dt^2} = -\frac{\partial E[\rho(\vec{r},t), \vec{R}_n]}{\partial \vec{R}_n}. \tag{6.50}$$

Here M_n and $\vec{R}_n$ are the masses and locations of the atomic nuclei.

As the solution of the full electronic problem at every time step is a very time consuming task we do not want to perform it all the time from scratch. Instead CPMD uses the previous values of the noninteracting electron wave functions $\{\phi_\mu\}$ of the DFT calculation (6.43) [don't confuse it with the Hartee-Fock orbitals!] and evolves them to the ground state for the current positions of the nuclei by an artificial molecular dynamics evolution. Hence both the nuclei $\{\vec{R}_n\}$ and the wave functions $\{\phi_\mu\}$ evolve in the same molecular dynamics scheme. The electronic degrees of freedoms are updated using an artificial dynamics:

$$m\frac{d^2\phi_\mu(\vec{r},t)}{dt^2} = -\frac{1}{2}\frac{\delta E[\rho(\vec{r},t), \vec{R}_n]}{\delta\phi_\mu^\dagger(\vec{r},t)} + \sum_\nu \Lambda_{\mu\nu}\phi_\nu(\vec{r},t), \tag{6.51}$$

where m is an artificial mass that needs to be chosen much lighter than the nuclear masses so that the electronic structure adapts quickly to the move of the nuclei. The Lagrange multipliers $\Lambda_{\mu\nu}$ need to be chose to ensure proper orthonormalization of the wave functions.

Since the exact form of the artifical dynamics of the electronic structure does not matter, we can evolve the expansion coefficients $d_{\mu n}$ of an expansion in terms of the basis functions as in equation (6.25) instead of evolving the wave functions. This gives the equations of motion

$$m\frac{d^2 d_{\mu n}}{dt^2} = -\frac{\partial E}{\partial d_{\mu n}} + \sum_\nu \Lambda_{\mu\nu} \sum_l S_{nl} d_{\nu l} \tag{6.52}$$

There are various algorithms to determine the $\Lambda_{\mu\nu}$ so that the wave functions stay orthonormal. We refer to text books and special lectures on CPMD for details.

6.10 Program packages

As the model Hamiltonian and the types of basis sets are essentially the same for all quantum chemistry applications flexible program packages have been written. There is thus usually no need to write your own programs – unless you want to implement a new algorithm.