Disclaimer

Open source projects are made available and contributed to under licenses that include terms that, for the protection of contributors, make clear that the projects are offered “as-is”, without warranty, and disclaiming liability for damages resulting from using the projects.

Citations

Development of this software depends on academic research grants. If you are using the package, please cite the following papers

[1] Long-range embedding of molecular ions and excitations in a polarizable molecular environment, Carl Poelking and Denis Andrienko


   *Phys. Rev. B*, 82, 193202, 2010

[5] Density-functional based determination of intermolecular charge transfer properties for large-scale morphologies, Björn Baumeier, James Kirkpatrick, and Denis Andrienko
   *Phys. Chem. Chem. Phys.* 12, 11103, 2010


[7] An approximate method for calculating transfer integrals based on the ZINDO Hamiltonian, James Kirkpatrick,

Copyright

VOTCA-CTP is free software. The entire package is available under the Apache License. For details, check the LICENSE file in the source code. The VOTCA-CTP source code is available on our homepage, www.votca.org.
Contents

1 Introduction .......................................................... 3

2 Theoretical background .............................................. 5
  2.1 Workflow .......................................................... 5
  2.2 Material morphology ............................................. 5
  2.3 Conjugated segments and rigid fragments ....................... 6
  2.4 Neighbor list ...................................................... 8
  2.5 Reorganization energy ............................................ 8
    2.5.1 Intramolecular reorganization energy ....................... 8
    2.5.2 Outersphere reorganization energy ......................... 9
  2.6 Site energies .................................................... 10
    2.6.1 Externally applied electric field .......................... 10
    2.6.2 Internal energy ............................................ 11
    2.6.3 Electrostatic interaction energy ........................... 11
    2.6.4 Induction energy - the Thole model ....................... 12
    2.6.5 Long-range Coulomb interactions .......................... 14
  2.7 Transfer integrals ................................................ 19
    2.7.1 Projection of monomer orbitals on dimer orbitals (DIPRO) 19
    2.7.2 DFT-based transfer integrals using DIPRO ................ 20
    2.7.3 ZINDO-based transfer integrals using MOO ............... 23
  2.8 Charge transfer rate ............................................ 24
    2.8.1 Classical charge transfer rate ............................ 24
    2.8.2 Semi-classical bimolecular rate ........................... 24
    2.8.3 Semi-classical rate ....................................... 25
  2.9 Master equation ................................................ 25
    2.9.1 Extrapolation to nondispersive mobilities .............. 26
  2.10 Macroscopic observables ...................................... 26
    2.10.1 Charge density ........................................... 27
    2.10.2 Current ................................................... 27
    2.10.3 Mobility and diffusion constant ........................ 27
    2.10.4 Spatial correlations of energetic disorder ............ 28
    2.10.5 DFT-based transfer integrals using DIPRO ............. 28

3 Input and output files ............................................. 31
  3.1 Atomistic topology ............................................. 31
  3.2 Mapping file .................................................... 33
  3.3 Conjugated segments ......................................... 34
  3.4 Molecular orbitals ............................................. 35
  3.5 Monomer calculations for DFT transfer integrals ............ 36
  3.6 Pair calculations for DFT transfer integrals ............... 37
  3.7 DFT transfer integrals ........................................ 39
  3.8 State file ....................................................... 40
<table>
<thead>
<tr>
<th>Section</th>
<th>Programs</th>
<th>Calculators</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Programs</td>
<td>Calculators</td>
</tr>
<tr>
<td>4.1.1</td>
<td>ctp_map</td>
<td>coupling</td>
</tr>
<tr>
<td>4.1.2</td>
<td>ctp_dump</td>
<td>log2mps</td>
</tr>
<tr>
<td>4.1.3</td>
<td>ctp_tools</td>
<td>molpol</td>
</tr>
<tr>
<td>4.1.4</td>
<td>ctp_run</td>
<td>pdb2map</td>
</tr>
<tr>
<td>4.1.5</td>
<td>ctp_parallel</td>
<td>pdb2top</td>
</tr>
<tr>
<td>4.1.6</td>
<td>moo_overlap</td>
<td>ptopreader</td>
</tr>
<tr>
<td>4.2</td>
<td>Calculators</td>
<td>Calculators</td>
</tr>
<tr>
<td>4.2.1</td>
<td>coupling</td>
<td>eanalyze</td>
</tr>
<tr>
<td>4.2.2</td>
<td>log2mps</td>
<td>eimport</td>
</tr>
<tr>
<td>4.2.3</td>
<td>molpol</td>
<td>izindo</td>
</tr>
<tr>
<td>4.2.4</td>
<td>pdb2map</td>
<td>jobwriter</td>
</tr>
<tr>
<td>4.2.5</td>
<td>pdb2top</td>
<td>neighborlist</td>
</tr>
<tr>
<td>4.2.6</td>
<td>ptopreader</td>
<td>profile</td>
</tr>
<tr>
<td>4.2.7</td>
<td>eanalyze</td>
<td>profile</td>
</tr>
<tr>
<td>4.2.8</td>
<td>eimport</td>
<td>rates</td>
</tr>
<tr>
<td>4.2.9</td>
<td>izindo</td>
<td>sandbox</td>
</tr>
<tr>
<td>4.2.10</td>
<td>multipole</td>
<td>xqmultipole</td>
</tr>
<tr>
<td>4.2.11</td>
<td>eoutersphere</td>
<td>edft</td>
</tr>
<tr>
<td>4.2.12</td>
<td>ewdbgpol</td>
<td>idft</td>
</tr>
<tr>
<td>4.2.13</td>
<td>ianalyze</td>
<td>pewald3d</td>
</tr>
<tr>
<td>4.2.14</td>
<td>iimport</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.15</td>
<td>izindo</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.16</td>
<td>jobwriter</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.17</td>
<td>neighborlist</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.18</td>
<td>pairdump</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.19</td>
<td>profile</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.20</td>
<td>rates</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.21</td>
<td>sandbox</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.22</td>
<td>stateserver</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.23</td>
<td>tdump</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.24</td>
<td>vaverage</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.25</td>
<td>zmultipole</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.26</td>
<td>edft</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.27</td>
<td>idft</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.28</td>
<td>pewald3d</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.29</td>
<td>qmmm</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.30</td>
<td>xqmultipole</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.31</td>
<td>energy2xml</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.32</td>
<td>integrals2xml</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.33</td>
<td>occupations2xml</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.34</td>
<td>pairs2xml</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.35</td>
<td>rates2xml</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.36</td>
<td>segments2xml</td>
<td>qmmm</td>
</tr>
<tr>
<td>4.2.37</td>
<td>trajectory2pdb</td>
<td>qmmm</td>
</tr>
</tbody>
</table>

**Bibliography**

67
Index

charge transfer rate, 22
  bimolecular, 22
  classical, 22
  semiclassical, 23
conjugated segment, 4
current, 25
  local, 25
diabatic states, 17
distributed multipoles, 9
electronic coupling, 17
  DFT, 18, 26
  ZINDO, 21
hopping site, 4
kinetic Monte Carlo, 23
Long-range interactions, 12
mobility, 25
neighbor list, 6
Pekar factor, 8
reorganization energy, 6
  intramolecular, 6
  outersphere, 7
rigid fragment, 5
site energy, 8
  distributed multipoles, 9
  Ewald, 12
  external field, 8
  internal, 9
  polarization, 10
  spatial correlation, 26
Thole model, 10
transfer integral, see electronic coupling
Chapter 1

Introduction

Charge carrier dynamics in an organic semiconductor can often be described in terms of charge hopping between localized states. The hopping rates depend on electronic coupling elements, reorganization energies, and site energies, which vary as a function of position and orientation of the molecules. The purpose of the VOTCA-CTP package [3] is to simplify the workflow for charge transport simulations, provide a uniform error-control for the methods, flexible platform for their development, and eventually allow in silico prescreening of organic semiconductors for specific applications.

The toolkit is implemented using modular concepts introduced earlier in the Versatile Object-oriented Toolkit for Coarse-graining Applications (VOTCA) [6]. It contains different programs, which execute specific tasks implemented in calculators representing an individual step in the workflow. Figure 1.1 summarizes a typical chain of commands to perform a charge transport simulation: First, the VOTCA code structures are adapted to reading atomistic trajectories, mapping them onto conjugated segments and rigid fragments, and substituting (if needed) rigid fragments with the optimized copies (ctp_map). The programs ctp_run and ctp_parallel (for heavy-duty tasks) are then used to calculate all bimolecular charge hopping rates (via precalculation of all required ingredients). Site energies (or energetic disorder) can be determined as a combination of internal (ionization potentials/electron affinities of single molecules) as well as electrostatic and polarization contributions within the molecular environment. The calculation of electronic coupling elements between conjugated segments from the corresponding molecular orbitals can be performed using a dimer-projection technique based on density-functional theory (DFT). This requires explicit calculations using quantum-chemistry software for which we provide interfaces to Gaussian, Turbomole, and NWChem. Alternatively, the molecular orbital overlap module calculates electronic coupling elements relying on the semi-empirical INDO Hamiltonian and molecular orbitals in the format provided by the Gaussian package.

The kinetic Monte Carlo module reads in the neighbor list, site coordinates, and hopping rates and performs charge dynamics simulations using either periodic boundary conditions or charge sources and sinks.

The toolkit is written as a combination of modular C++ code and scripts. The data transfer between programs is implemented via a state file (sql database), which is also used to restart simulations. Analysis functions and most of the calculation routines are encapsulated by using the observer pattern [8] which allows the implementation of new functions as individual modules.

In the following chapter 2, we summarize the theoretical background of the workflow of charge transport simulations and in particular its individual steps. Chapter 3 describes the structure and content of input and output files, while a full reference of programs and calculators is available in chapter 4. For a hands-on tutorial, the reader is referred to the VOTCA-CTP project page at http://code.google.com/p/votca-ctp/.
CHAPTER 1. INTRODUCTION

Input files:
- conf.gro
- GROMACS trajectory
- topol.tpr
- GROMACS topology
- map.xml
- mapping and energies
- options.xml
- options for calculators

Output files:
- state.sql
- sqlite3 database file for data transfer between modules

Mapping, generation of the sql database
Converts and partitions atomistic GROMACS trajectory

\`
ctp_map -t topol.tpr -c traj.xtc -s map.xml -f state.sql
\`

Useful tools:
- toolpdb2map

Neighbor list
Indicates close molecular pairs between which charge transfer rates will be calculated

\`
ctp_run -o options.xml -f state.sql -e neighborlist
\`

Site energies
Calculates electrostatic and polarization contribution to site energies

\`
ctp_run -o options.xml -f state.sql -e multipole
\`

Internal site and reorganization energies
Imports internal site energy (IP, EA) and reorganization energies for charging and discharging to state.sql

\`
ctp_run -o options.xml -f state.sql -e internal
\`

Transfer integrals

Transfer integrals with DFT
Calculate the relevant transport orbitals of monomers

\`
ctp_parallel -o options.xml -f state.sql -e edft -j "write run"
\`

Transfer integrals with ZINDO
Calculate electronic coupling elements for all pairs in the neighbor list

\`
ctp_run -o options.xml -f state.sql -e izindo
\`

One can choose between quantum-chemical (computationally expensive) or semi-empirical (fast, but not always sufficiently accurate) evaluation of transfer integrals.

DFT

Monomers with DFT
Calculate the relevant transport orbitals of monomers

\`
ctp_parallel -o options.xml -f state.sql -e edft -j "write run"
\`

Outersphere reorganization energies
Contribution to reorganization of surrounding molecules due to polarization. (optional for Marcus rates)

\`
ctp_run -o options.xml -f state.sql -e outersphere
\`

Charge transfer rates
Calculates rates for charge transfer among all pairs in the neighbor list

\`
ctp_run -o options.xml -f state.sql -e rates
\`

Charge dynamics via kMC
Hopping of charge carriers simulated via kinetic Monte Carlo

\`
kmc_run -o options.xml -f state.sql -e kmcmultiple
\`

Get list of available calculators: \texttt{ctp_run/ctp_parallel/kmc_run -l}
Get help and list of options for a calculator: \texttt{ctp_run/ctp_parallel/kmc_run -d neighborlist}

Figure 1.1: A practical workflow of charge transport simulations using VOTCA-CTP. The theoretical background of the individual steps is given in chapter 2. Chapter 3 describes the content of input and output files, while a full reference of programs and calculators is available in chapter 4.
Chapter 2

Theoretical background

2.1 Workflow

A typical workflow of charge transport simulations is depicted in figure 2.1. The first step is the simulation of an atomistic morphology, which is then partitioned on hopping sites. The coordinates of the hopping sites are used to construct a list of pairs of molecules, or neighbor list.

For each pair an electronic coupling element, a reorganization energy, a driving force, and eventually the hopping rate are evaluated. The neighbor list and hopping rates define a directed graph. The corresponding master equation is solved using the kinetic Monte Carlo method, which allows to explicitly monitor the charge dynamics in the system as well as to calculate time or ensemble averages of occupation probabilities, charge fluxes, correlation functions, and field-dependent mobilities.

2.2 Material morphology

There is no generic recipe on how to predict a large-scale atomistically-resolved morphology of an organic semiconductor. The required methods are system-specific: for ultra-pure crystals, for
example, density-functional methods can be used provided the crystal structure is known from experiment. For partially disordered organic semiconductors, however, system sizes much larger than a unit cell are required. Classical molecular dynamics or Monte Carlo techniques are then the methods of choice.

In molecular dynamics, atoms are represented by point masses which interact via empirical potentials prescribed by a force-field. Force-fields are parametrized for a limited set of compounds and their refinement is often required for new molecules. In particular, special attention shall be paid to torsion potentials between successive repeat units of conjugated polymers or between functional groups and the $\pi$-conjugated system. First-principles methods can be used to characterize the missing terms of the potential energy function.

Self-assembling materials, such as soluble oligomers, discotic liquid crystals, block copolymers, partially crystalline polymers, etc., are the most complicated to study. The morphology of such systems often has several characteristic length scales and can be kinetically arrested in a thermodynamically non-equilibrium state. For such systems, the time- and length-scales of atomistic simulations might be insufficient to equilibrate or sample desired morphologies. In this case, systematic coarse-graining can be used to enhance sampling [6]. Note that the coarse-grained representation must reflect the structure of the atomistic system and allow for back-mapping to the atomistic resolution.

Here we assume that the morphology is already known, that is we know how the topology and the coordinates of all atoms in the systems at a given time. VOTCA-CTP can read standard GROMACS topology files. Custom definitions of atomistic topology via XML files are also possible. Since the description of the atomistic topology is the first step in the charge transport simulations, it is important to follow simple conventions on how the system is partitioned on molecules, residues, and how atoms are named in the topology. Required input files are described in section atomistic topology.

### 2.3 Conjugated segments and rigid fragments

With the morphology at hand, the next step is partitioning the system on hopping sites, or conjugated segments, and calculating charge transfer rates between them. Physically intuitive arguments can be used for the partitioning, which reflects the localization of the wave function of a charge. For most organic semiconductors, the molecular architecture includes relatively rigid, planar $\pi$-conjugated systems, which we will refer to as rigid fragments. A conjugated segment can contain one or more of such rigid fragments, which are linked by bonded degrees of freedom.
The dynamics of these degrees of freedom evolves on timescales much slower than the frequency of the internal promoting mode. In some cases, e.g. glasses, it can be ‘frozen’ due to non-bonded interactions with the surrounding molecules.

To illustrate the concept of conjugated segments and rigid fragments, three representative molecular architectures are shown in figure 2.2. The first one is a typical discotic liquid crystal, hexabenzocoronene. It consists of a conjugated core to which side chains are attached to aid self-assembly and solution processing. In this case the orbitals localized on side chains do not participate in charge transport and the conjugated \( \pi \)-system is both, a rigid fragment and a conjugated segment. In \( \text{Alq}_3 \), a metal-coordinated compound, a charge carrier is delocalized over all three ligands. Hence, the whole molecule is one conjugated segment. Individual ligands are relatively rigid, while energies of the order of \( k_B T \) are sufficient to reorient them with respect to each other. Thus the Al atom and the three ligands are rigid fragments. In the case of a conjugated polymer, one molecule can consist of several conjugated segments, while each backbone repeat unit is a rigid fragment. Since the conjugation along the backbone can be broken due to large out-of-plane twists between two repeat units, an empirical criterion, based on the dihedral angle, can be used to partition the backbone on conjugated segments [9]. However, such intuitive partitioning is, to some extent, arbitrary and shall be validated by other methods [10–12].

After partitioning, an additional step is often required to remove bond length fluctuations introduced by molecular dynamics simulations, since they are already integrated out in the derivation of the rate expression. This is achieved by substituting respective molecular fragments with rigid, planar \( \pi \)-systems optimized using first-principles methods. Centers of mass and gyration tensors are used to align rigid fragments, though a custom definition of local axes is also possible. Such a procedure also minimizes discrepancies between the force-field and first-principles-based ground state geometries of conjugated segments, which might be important for calculations of electronic couplings, reorganization energies, and intramolecular driving forces.

To partition the system on hopping sites and substitute rigid fragments with the corresponding ground-state geometries \texttt{ctp_map} program is used:

```
Mapping the GROMACS trajectory

\texttt{ctp_map -t topol.tpr -c traj.xtc -s map.xml -f state.sql}
```

It reads in the GROMACS topology (\texttt{topol.tpr}) and trajectory (\texttt{traj.xtc}) files, definitions of conjugated segments and rigid fragments (\texttt{map.xml}) and outputs coordinates of conjugated segments (hopping sites) and rigid fragments (as provided in the MD trajectory and after rigidification) to the state file (\texttt{state.sql}). In order to do this, a mapping file \texttt{map.xml} has to be provided, which specifies the corresponding atoms in the different representations. After this step, all information (frame number, dimensions of the simulation box, etc) are stored in the state file and only this file is used for further calculations.

**Attention**

VOTCA-CTP requires a wrapped trajectory for mapping the segments and fragments, so all molecules should be whole in the frame.

In order to visually check the mapping one can use either the \texttt{t.dump} calculator or the program \texttt{ctp_dump} with the calculator \texttt{trajectory2pdb}.

```
Writing a mapped trajectory with \texttt{ctp_dump}

\texttt{ctp_dump -f state.sql -e trajectory2pdb}
```
It reads in the state file created by \texttt{ctp\_map} and outputs two trajectory files corresponding to the original and rigidified atom coordinates. To check the mapping, it is useful to superimpose the three outputs (original atomistic, atomistic stored in the state file, and rigidified according to ground state geometries), e.g., with \texttt{VMD}.

Writing a mapped trajectory with \texttt{tdump}

\begin{verbatim}
 ctp\_run -f state.sql -o options.xml -e tdump
\end{verbatim}

It also reads in the state file but appends the coordinates to a \texttt{pdb} file. So make sure to delete old \texttt{QM.pdb} and \texttt{MD.pdb} if you want to create a new image.

2.4 Neighbor list

A list of neighboring conjugated segments, or neighbor list, contains all pairs of conjugated segments for which coupling elements, reorganization energies, site energy differences, and rates are evaluated. Two segments are added to this list if the distance between centers of mass of any of their rigid fragments is below a certain cutoff. This allows neighbors to be selected on a criterion of minimum distance of approach rather than center of mass distance, which is useful for molecules with anisotropic shapes. The neighbor list can be generated from the atomistic trajectory by using the \texttt{neighborlist} calculator. This calculator requires a cutoff, which can be specified in the \texttt{options.xml} file. The list is saved to the \texttt{state.sql} file:

Generating a neighbor list

\begin{verbatim}
 ctp\_run -o options.xml -f state.sql -e neighborlist
\end{verbatim}

2.5 Reorganization energy

The reorganization energy $\lambda_{ij}$ takes into account the change in nuclear (and dielectric) degrees of freedom as the charge moves from donor $i$ to acceptor $j$. It has two contributions: intramolecular, $\lambda_{ij}^{\text{int}}$, which is due to reorganization of nuclear coordinates of the two molecules forming the charge transfer complex, and intermolecular (outersphere), $\lambda_{ij}^{\text{out}}$, which is due to the relaxation of the nuclear coordinates of the environment. In what follows we discuss how these contributions can be calculated.

2.5.1 Intramolecular reorganization energy

If intramolecular vibrational modes of the two molecules are treated classically, the rearrangement of their nuclear coordinates after charge transfer results in the dissipation of the internal reorganization energy, $\lambda_{ij}^{\text{int}}$. It can be computed from four points on the potential energy surfaces (PES) of both molecules in neutral and charged states, as indicated in figure 2.3.

Adding the contributions due to discharging of molecule $i$ and charging of molecule $j$ yields [13]

\begin{equation}
\lambda_{ij}^{\text{int}} = \lambda_{ij}^{\text{en}} + \lambda_{ij}^{\text{nc}} = U_{ij}^{\text{nC}} - U_{ij}^{\text{CN}} + U_{ij}^{\text{CoN}} - U_{ij}^{\text{C}}. \tag{2.1}
\end{equation}

Here $U_{ij}^{\text{nC}}$ is the internal energy of the neutral molecule $i$ in the geometry of its charged state (small $n$ denotes the state and capital $C$ the geometry). Similarly, $U_{ij}^{\text{CoN}}$ is the energy of the charged molecule $j$ in the geometry of its neutral state. Note that the PES of the donor and acceptor are not identical for chemically different compounds or for conformers of the same molecule. In this
2.5. REORGANIZATION ENERGY

Figure 2.3: Potential energy surfaces of (a) donor and (b) acceptor in charged and neutral states. After the change of the charge state both molecules relax their nuclear coordinates. If all vibrational modes are treated classically, the total internal reorganization energy and the internal energy difference of the electron transfer reaction are

\[ \lambda_{ij}^{\text{int}} = \lambda_i^{cn} + \lambda_j^{nc} \]

and

\[ \Delta E_{ij}^{\text{int}} = \Delta U_i - \Delta U_j \]

respectively.

Intramolecular reorganization energies for discharging (\(\lambda_i^{cn}\)) and charging (\(\lambda_j^{nc}\)) of a molecule need to be determined using quantum-chemistry and given in map.xml. The values are written to the state.sql using the calculator einternal (see also internal energy):

```
ctp_run -o options.xml -f state.sql -e einternal
```

2.5.2 Outersphere reorganization energy

During the charge transfer reaction, also the molecules outside the charge transfer complex reorient and polarize in order to adjust for changes in electric potential, resulting in the outersphere contribution to the reorganization energy. \(\lambda_{ij}^{\text{out}}\) is particularly important if charge transfer occurs in a polarizable environment. Assuming that charge transfer is much slower than electronic polarization but much faster than nuclear rearrangement of the environment, \(\lambda_{ij}^{\text{out}}\) can be calculated from the electric displacement fields created by the charge transfer complex [14]

\[ \lambda_{ij}^{\text{out}} = \frac{c_p}{2\epsilon_0} \int_{V_{\text{out}}} dV \left[ \vec{D}_I - \vec{D}_F \right]^2, \]  

(2.2)  
equ:lambda_outer1

where \(\epsilon_0\) is the permittivity of free space, \(\vec{D}_I, F(\vec{r})\) are the electric displacement fields created by the charge transfer complex in the initial (charge on molecule \(i\)) and final (charge transferred to molecule \(j\)) states, \(V_{\text{out}}\) is the volume outside the complex, and \(c_p = \frac{1}{\epsilon_{\text{opt}}} - \frac{1}{\epsilon_{\text{s}}}\) is the Pekar factor, which is determined by the low (\(\epsilon_{\text{s}}\)) and high (\(\epsilon_{\text{opt}}\)) frequency dielectric permittivities.

Eq. (2.2) can be simplified by assuming spherically symmetric charge distributions on molecules \(i\) and \(j\) with total charge \(\epsilon\). Integration over the volume \(V_{\text{out}}\) outside of the two spheres of radii \(R_i\) and \(R_j\) centered on molecules \(i\) and \(j\) leads to the classical Marcus expression for the outersphere reorganization energy

\[ \lambda_{ij}^{\text{out}} = \frac{c_p e^2}{4\pi\epsilon_0} \left( \frac{1}{2R_i} + \frac{1}{2R_j} - \frac{1}{r_{ij}} \right), \]  

(2.3)  
equ:lambda_outer2

where \(r_{ij}\) is the molecular separation. While eq. (2.3) captures the main physics, e.g. predicts smaller outer-sphere reorganization energies (higher rates) for molecules at smaller separations,
it often cannot provide quantitative estimates, since charge distributions are rarely spherically symmetric.

Alternatively, the displacement fields can be constructed using the atomic partial charges. The difference of the displacement fields at the position of an atom \( b_k \) outside the charge transfer complex (molecule \( k \neq i,j \)) can be expressed as

\[
\bar{D}_i(\bar{r}_{b_k}) - \bar{D}_j(\bar{r}_{b_k}) = \sum_{a_i} \frac{q_{a_i}^c - q_{a_i}^d}{4\pi \left| \bar{r}_{a_i} - \bar{r}_{b_k} \right|^3} + \frac{q_{a_j}^c - q_{a_j}^d}{4\pi \left| \bar{r}_{a_j} - \bar{r}_{b_k} \right|^3},
\]

(2.4)

where \( q_{a_i}^c \) (\( q_{a_j}^d \)) is the partial charge of atom \( a \) of the neutral (charged) molecule \( i \) in vacuum. The partial charges of neutral and charged molecules are obtained by fitting their values to reproduce the electrostatic potential of a single molecule (charged or neutral) in vacuum. Assuming a uniform density of atoms, the integration in eq. (2.2) can be rewritten as a density-weighted sum over all atoms excluding those of the charge transfer complex.

The remaining unknown needed to calculate \( \lambda_{ij}^{\text{out}} \) is the Pekar factor, \( c_p \). In polar solvents \( \epsilon_s \gg \epsilon_{\text{opt}} \sim 1 \) and \( c_p \) is of the order of 1. In most organic semiconductors, however, molecular orientations are fixed and therefore the low frequency dielectric permittivity is of the same order of magnitude as \( \epsilon_{\text{opt}} \). Hence, \( c_p \) is small and its value is very sensitive to differences in the permittivities.

Outersphere reorganization energies for all pairs of molecules in the neighbor list can be computed from the atomistic trajectory by using the outersphere calculator.

Two methods can be used to compute \( \lambda_{ij}^{\text{out}} \). The first method uses the atomistic partial charges of neutral and charged molecules from files specified in \( \text{map.xml} \) and eq. (2.2). The Pekar factor \( c_p \) and a cutoff radius based on molecular centers of mass have to be specified in the \( \text{options.xml} \) file.

If this method is computationally prohibitive, \( \lambda_{ij}^{\text{out}} \) can be computed using eq. (2.3), which assumes spherical charge distributions on the molecules. In this case the radii of these spheres are specified in \( \text{segments.xml} \), while the Pekar factor \( c_p \) is given in the \( \text{options.xml} \) file and no cutoff radius is needed.

The outer sphere reorganization energies are saved to the \( \text{state.sql} \) file:

```
Outersphere reorganization energy
ctp_run -o options.xml -f state.sql -e outersphere
```

### 2.6 Site energies

A charge transfer reaction between molecules \( i \) and \( j \) is driven by the site energy difference, \( \Delta E_{ij} = E_i - E_j \). Since the transfer rate, \( \omega_{ij} \), depends exponentially on \( \Delta E_{ij} \) (see eq. (2.31)) it is important to compute its distribution as accurately as possible. The total site energy difference has contributions due to externally applied electric field, electrostatic interactions, polarization effects, and internal energy differences. In what follows we discuss how to estimate these contributions by making use of first-principles calculations and polarizable force-fields.

#### 2.6.1 Externally applied electric field

The contribution to the total site energy difference due to an external electric field \( \bar{F} \) is given by

\[
\Delta E_{ij}^{\text{ext}} = q \bar{F} \cdot \bar{r}_{ij},
\]

where \( q = \pm e \) is the charge and \( \bar{r}_{ij} = \bar{r}_i - \bar{r}_j \) is a vector connecting molecules \( i \) and \( j \). For typical distances between small molecules, which are of the order of 1 nm, and moderate fields of \( F < 10^8 \text{V/m} \) this term is always smaller than 0.1 eV.
2.6. SITE ENERGIES

2.6.2 Internal energy

The contribution to the site energy difference due to different internal energies (see figure 2.3) can be written as

$$\Delta E^{\text{int}}_{ij} = \Delta U_i - \Delta U_j = (U^{\text{nc}}_i - U^{\text{nc}}_j) - (U^{\text{nc}}_j - U^{\text{nc}}_j), \quad (2.5)$$

where $U^{\text{nc}}(nN)$ is the total energy of molecule $i$ in the charged (neutral) state and geometry. $\Delta U_i$ corresponds to the adiabatic ionization potential (or electron affinity) of molecule $i$, as shown in figure 2.3. For one-component systems and negligible conformational changes $\Delta E^{\text{int}}_{ij} = 0$, while it is significant for donor-acceptor systems.

Internal energies determined using quantum-chemistry need to be specified in `map.xml`. The values are written to the `state.sql` using the calculator `einormal` (see also intramolecular reorganization energy):

```
# Internal energies
ctp_run -o options.xml  -f state.sql -e einormal
```

2.6.3 Electrostatic interaction energy

We represent the molecular charge density by choosing multiple expansion sites (“polar sites”) per molecule in such a way as to accurately reproduce the molecular electrostatic potential (ESP), with a set of suitably chosen multipole moments $\{Q_{lk}^a\}$ (in spherical-tensor notation) allocated to each site. The expression for the electrostatic interaction energy between two molecules $A$ and $B$ in the multi-point expansion includes an implicit sum over expansion sites $a \in A$ and $b \in B$.

$$U_{AB} = \sum_{a \in A} \sum_{b \in B} \hat{Q}_{l_1,k_1}^{a} T_{l_1,k_1,l_2,k_2}^{a,b} \hat{Q}_{l_2,k_2}^{b} \equiv \hat{Q}_{l_1,k_1}^{a} T_{l_1,k_1,l_2,k_2}^{a,b} \hat{Q}_{l_2,k_2}^{b}, \quad (2.6)$$

where we have used the Einstein sum convention for the site indices $a$ and $b$ on the right-hand side of the equation, in addition to the sum convention that is in place for the multipole-moment components $l \equiv l_1 k_1$ and $u \equiv l_2 k_2$. The $T_{l_1,k_1,l_2,k_2}^{a,b}$ are tensors that mediate the interaction between a multipole component $l_1 k_1$ on site $a$ with the moment $l_2 k_2$ on site $b$. If we include the molecular environment into a perturbative term $W$ to enter in the single-molecule Hamiltonian, the above expression is exactly the first-order correction to the energy where the quantum-mechanical detail has been absorbed in classical multipole moments.

The are a number of strategies how to arrive at such a collection of distributed multipoles. They can be classified according to whether the multipoles are derived (a) from the electrostatic potential generated by the SCF charge density or (b) from a decomposition of the wavefunction itself. Here, we will only draft two of those approaches, CHELPG [15] from category (a) and DMA [16] from category (b).

The CHELPG (CHarges from ELectrostatic Potentials, Grid-based) method relies on performing a least-squares fit of atom-placed charges to reproduce the electrostatic potential as evaluated from the SCF density on a regularly spaced grid [15]. The fitted charges result from minimizing the Lagrangian function [17]:

$$z(\{q_i\}) = \sum_{k=1}^{M} \phi(\vec{r}_k) - \sum_{i=1}^{N} \frac{1}{\pi \varepsilon_0} \frac{q_i}{|\vec{r}_i - \vec{r}_k|} + \lambda \left( q_{\text{mol}} - \sum_{i=1}^{N} q_i \right), \quad (2.7)$$

with $M$ grid points, $N$ atomic sites, the set of atomic partial charges $\{q_i\}$ and the SCF potential $\phi$. The Lagrange multiplier $\lambda$ constrains the sum of the fitted charges to the molecular charge $q_{\text{mol}}$. The main difference from other fitting schemes [18] is the algorithm that selects the positions...
CHAPTER 2. THEORETICAL BACKGROUND

2.6.4 Induction energy - the Thole model

If we in addition to the permanent set of multipole moments \( \{ Q_{at} \} \) allow for induced moments \( \{ \Delta Q_{at} \} \) and penalize their generation with a bilinear form (giving rise to a strictly positive contribution to the energy),

\[
U_{\text{int}} = \frac{1}{2} \sum_{A} \Delta Q_{at}^{a} \eta_{tt}^{aa} \Delta Q_{ct}^{a},
\]

(2.11)
2.6. SITE ENERGIES

It can be shown that the induction contribution to the site energy evaluates to an expression where all interactions between induced moments have cancelled out, and interactions between permanent and induced moments are scaled down by $1/2$ [20]:

$$U_{pu} = \frac{1}{2} \sum_A \sum_B \sum_{A \neq B} \left[ \Delta Q_{tu}^{aa} T_{tu}^{ab} Q_{u}^{b} + \Delta Q_{tu}^{ab} T_{tu}^{ab} Q_{u}^{a} \right].$$  \hspace{1cm} (2.12)  

This term can be viewed as the second-order (induction) correction to the molecular interaction energy. The sets of $\{Q_{t}^{a}\}$ are solved for self-consistently via

$$\Delta Q_{t}^{a} = -\sum_{B \neq A} \alpha_{tt}^{aa} T_{tu} Q_{u}^{b} (Q_{u}^{b} + \Delta Q_{u}^{b}),$$  \hspace{1cm} (2.13)  

where the polarizability tensors $\alpha_{tt}^{aa}$ are given by the inverse of $\eta_{tt}^{aa}$.

With eqs. 2.13 and 2.12 we have at hand expressions that allow us to compute the induction energy contribution to site energies in an iterative manner based on a set of molecular distributed multipoles $\{Q_{t}^{a}\}$ and polarizabilities $\{\alpha_{tt}^{aa}\}$. We have drafted in the previous section how to obtain the former from a wavefunction decomposition or fitting scheme (GDMA, CHELPG). The $\{\alpha_{tt}^{aa}\}$ can be derived formally (or rather: read off) from a perturbative expansion of the molecular interaction. In this work we make use of the Thole model [21, 22] as a semi-empirical approach to obtain the sought-after point polarizabilities in the local dipole approximation, that is,

$$\alpha_{tt}^{aa} = \alpha_{tt}^{aa} (1 + |\vec{u}|)^{2} \delta_{\alpha \beta},$$

where the specific form of $\alpha_{tt}^{aa}$ references the dipole-moment component.

The Thole model is based on a modified dipole-dipole interaction, which can be reformulated in terms of the interaction of smeared charge densities. This has been shown to be necessary due to the divergent head-to-tail dipole-dipole interaction that otherwise results at small interseparations on the Å scale [21–23]. Smearing out the charge distribution mimics the nature of the QM wavefunction, which effectively guards against this unphysical polarization catastrophe. Since the point dipoles however only react individually to the external field, any correlation effects as were still accounted for in the $\{\alpha_{tt}^{aa}\}$ are lost, except perhaps those correlations that are due to the mere classical field interaction.

The smearing of the nuclei-centered multipoles is obtained via a fractional charge density $\rho_{f}(\vec{u})$ which should be normalized to unity and fall off rapidly as of a certain radius $\vec{u} = \vec{u}(\vec{R})$. The latter is related to the physical distance vector $\vec{R}$ connecting two interacting sites via a linear scaling factor that takes into account the magnitude of the isotropic site polarizabilities $\alpha^{a}$. This isotropic fractional charge density gives rise to a modified potential

$$\phi(\vec{u}) = -\frac{1}{4\pi\varepsilon_{0}} \int_{0}^{u} 4\pi u'\rho(u')du'.$$  \hspace{1cm} (2.14)  

We can relate the multipole interaction tensor $T_{ij...}$ (this time in Cartesian coordinates) to the fractional charge density in two steps: First, we rewrite the tensor in terms of the scaled distance vector $\vec{u}$,

$$T_{ij...}(\vec{R}) = f(\alpha^{a}\alpha^{b}) t_{ij...}(\vec{u}(\vec{R}, \alpha^{a}\alpha^{b})),$$  \hspace{1cm} (2.15)  

where the specific form of $f(\alpha^{a}\alpha^{b})$ results from the choice of $u(\vec{R}, \alpha^{a}\alpha^{b})$. Second, we demand that the smeared interaction tensor $t_{ij...}$ is given as usual by the appropriate derivative of the potential in eq. 2.14,

$$t_{ij...}(\vec{u}) = -\partial_{ui_{i}}\partial_{uj_{j}}...\phi(\vec{u}).$$  \hspace{1cm} (2.16)  

It turns out that for a suitable choice of $\rho_{f}(\vec{u})$, the modified interaction tensors can be rewritten in such a way that powers $n$ of the distance $R = |\vec{R}|$ are damped with a damping function $\lambda_{n}(\vec{u}(\vec{R}))$ [24].
There is a large number of fractional charge densities $\rho_f(\vec{u})$ that have been tested for the purpose of giving best results for the molecular polarizability as well as interaction energies. Note how a great advantage of the Thole model is the exceptional transferability of the atomic polarizabilities to compounds not used for the fitting procedure [22]. In fact, for most organic molecules, a fixed set of atomic polarizabilities ($\alpha_C = 1.334, \alpha_H = 0.496, \alpha_N = 1.073, \alpha_O = 0.873, \alpha_S = 2.926 \text{ Å}^3$) based on atomic elements yields satisfactory results.

VOTCA implements the Thole model with an exponentially-decaying fractional charge density

$$\rho(\vec{u}) = \frac{3a}{4\pi} \exp(-au^3),$$

(2.17)

where $\vec{u}(\vec{R}, \alpha^a \alpha^b) = \vec{R}/(\alpha^a \alpha^b)^{1/6}$ and the smearing exponent $a = 0.39$ (which can however be changed from the program options), as used in the AMOEBA force field [24].

Even though the Thole model performs very well for many organic compounds with only the above small set of element-based polarizabilities, conjugated molecules may require a more intricate parametrization. The simplest approach is to resort to scaled polarizabilities to match the effective molecular polarizable volume $V \sim \alpha_x \alpha_y \alpha_z$ as predicted by QM calculations (here $\alpha_x, \alpha_y, \alpha_z$ are the eigenvalues of the molecular polarizability tensor). The molpol tool assists with this task, it self-consistently calculates the Thole polarizability for an input mps-file and optimizes (if desired) the atomic polarizabilities in the above simple manner.

**Generate Thole-type polarizabilities for a segment**

```
ctp_tools -o options.xml -e molpol
```

The electrostatic and induction contribution to the site energy is evaluated by the `emultipole` calculator. Atomistic partial charges for charged and neutral molecules are taken from mps-files (extended GDMA format) specified in map.xml. Note that, in order to speed up calculations for both methods, a cut-off radius (for the molecular centers of mass) can be given in options.xml. Threaded execution is advised.

**Electrostatic and induction corrections**

```
ctp_run -o options.xml -f state.sql -e emultipole
```

Furthermore available are `zmultipole`, which extends `emultipole` to allow for an electrostatic buffer layer (loosely related to the z-buffer in OpenGL, hence the name) and anisotropic point polarizabilities. For the interaction energy of charged clusters of any user-defined composition (Frenkel states, CT states, ...), `xqmultipole` can be used.

**Interaction energy of charged molecular clusters embedded in a molecular environment**

```
ctp_parallel -o options.xml -f state.sql -e xqmultipole
```

### 2.6.5 Long-range Coulomb interactions

This section is a practical guide for doing electrostatic calculations in slabs using aperiodic Ewald scheme described in [1].

First, you will need to generate the required quantum mechanical reference, comprising molecular charge density and polarizability for all charge states (neutral, cationic, anionic). For example, you can use GAUSSIAN to do this:

```plaintext
... #p b3lyp/6-31+g(d,p) pop(full,chelpg) polar(dipole) nosymm test ...
```

Afterwards, you have to generate all required \textit{mps}-files with distributed multipoles and polarizabilities. The options file should point to the QM log-files, as well as contain the target molecular polarizability tensor in upper-diagonal order $xx \text{ } xy \text{ } xz \text{ } yy \text{ } yz \text{ } zz$ in units of Å$^3$.

\begin{verbatim}
$ ctp_tools -e log2mps -o options.xml
$ ctp_tools -e molpol -o options.xml

<options>
  <log2mps>
    <package>gaussian</package>
    <logfile>input.log</logfile>
    <mpsfile></mpsfile>
  </log2mps>
  <molpol>
    <mpsfiles>
      <input>input.mps</input>
      <output>output.mps</output>
      <polar>output.xml</polar>
    </mpsfiles>
    <induction>
      <expdamp>0.39000</expdamp>
      <wSOR>0.30000</wSOR>
      <maxiter>1024</maxiter>
      <tolerance>0.00001</tolerance>
    </induction>
    <target>
      <optimize>true</optimize>
      <molpol>77 0 0 77 0 77</molpol>
      <tolerance>0.00001</tolerance>
    </target>
  </molpol>
</options>
\end{verbatim}

Next, generate the \textit{mps} table, which relates the state of the molecule (neutral, cation, anion) with a corresponding electrostatic representation. This is provided by the \textit{stateserver}. The resulting output file has the default name “mps.tab”:

\begin{verbatim}
$ ctp_run -e stateserver -o options.xml -f state.sql

<options>
  <stateserver>
    <keys>mps</keys>
  </stateserver>
</options>
\end{verbatim}

Next, generate the job file. This job file lists the electrostatic configurations that are to be investigated. It can either be composed by hand or generated automatically from the sql file via the \textit{jobwriter}, which at present takes either “mps.chrg”, “mps.single” or “mps.ct” as key, where the latter resorts to a neighbor list. The resulting xml-file has the default name “jobwriter.mps.chrg.xml”.

\begin{verbatim}
$ ctp_run -e jobwriter -o options.xml -f state.sql

<options>
  <jobwriter>
    <keys>mps.chrg</keys>
  </jobwriter>
</options>
\end{verbatim}
The input string in the job file for the long-range corrected calculators has the same format as for the \textit{xqmultipole} calculator, "id1:name1:mps1 id2:name2:mps2 ...". See sample below.

\begin{verbatim}
<jobs>
  <job>
    <id>1</id>
    <tag>1e:2h</tag>
    <input>1:spl:MP_FILES/spl_e.mps 2:spl:MP_FILES/spl_h.mps</input>
    <status>AVAILABLE</status>
  </job>
  <job>
    ...
  </job>
  ...
</jobs>
\end{verbatim}

Next, generate the \textit{ptop}-file that stores the induction state of the neutral background. The responsible \textit{ewdbgpol} calculator can be run in a threaded fashion, depending on system size. The resulting \textit{ptop}-file has the default name "bgp_main.ptop".

\$ ctp\_run -e ewdbgpol -o options.xml -f state.sql -t 8

\begin{verbatim}
<options>
  <ewdbgpol>
    <multipoles>system.xml</multipoles>
    <control>
      <mps_table>mps.tab</mps_table>
      <pdb_check>1</pdb_check>
    </control>
    <coulombmethod>
      <method>ewald</method>
      <cutoff>6</cutoff>
      <shape>xyslab</shape>
    </coulombmethod>
    <polarmethod>
      <method>thole</method>
      <wSOR_N>0.350</wSOR_N>
      <aDamp>0.390</aDamp>
    </polarmethod>
    <convergence>
      <energy>1e-05</energy>
      <kfactor>100</kfactor>
      <rfactor>6</rfactor>
    </convergence>
  </ewdbgpol>
</options>
\end{verbatim}

Finally, run the energy computation using \textit{pewald3d}. This job calculator is wrapped by the \texttt{ctp\_parallel} executable, which allows for communication between different processes via the job and state file. Unfortunately, communication, though guarded by a file lock, may fail on some architectures in the event of frequent accesses to the job file. This frequency can be controlled by the \texttt{-c/-cache}
argument, which defines the number of jobs that are loaded in one batch by a specific process/node.

```bash
$ ctp_parallel -e pewald3d -o options.xml -f /absolute/path/to/state.sql -s 0 -t 8 -o
```

```xml
<options>
  <ewald>
    <jobcontrol/>
      <job_file>/absolute/path/to/jobs.xml</job_file>
    </jobcontrol>
    <multipoles>
      <mapping>system.xml</mapping>
      <mps_table>mps.tab</mps_table>
      <polar_bg>bgp_main.ptop</polar_bg>
      <pdb_check>0</pdb_check>
    </multipoles>
    <coulombmethod>
      <method>ewald</method>
      <cutoff>8</cutoff>
      <shape>xyslab</shape>
      <save_nblist>false</save_nblist>
    </coulombmethod>
    <polarmethod>
      <method>thole</method>
      <induce>1</induce>
      <cutoff>4</cutoff>
      <tolerance>0.001</tolerance>
      <radial_dielectric>4.0</radial_dielectric>
    </polarmethod>
    <tasks>
      <calculate_fields>true</calculate_fields>
      <polarize_fg>true</polarize_fg>
      <evaluate_energy>true</evaluate_energy>
      <apply_radial>false</apply_radial>
    </tasks>
    <coarsegrain>
      <cg_background>false</cg_background>
      <cg_foreground>false</cg_foreground>
      <cg_radius>3</cg_radius>
      <cg_anisotropic>true</cg_anisotropic>
    </coarsegrain>
    <convergence>
      <energy>1e-05</energy>
      <kfactor>100</kfactor>
      <rfactor>6</rfactor>
    </convergence>
  </ewald>
</options>
```

Parse the output. The results from the computation are stored in the same job file that was supplied to the calculator. The key data is provided in the output/summary section and consists of the electrostatic and induction contributions output/summary/eindu and output/summary/estat. Note that only configuration energy differences carry meaning. The parsing is best done by script, as the "-j/-jobs read" option for pewald3d is not yet implemented.
CHAPTER 2. THEORETICAL BACKGROUND

<jobs>
  <job>
    <id>1</id>
    <tag>le:2h</tag>
    <input>1:sp1:MP_FILES/sp1_e.mps 2:sp1:MP_FILES/sp1_h.mps</input>
    <status>COMPLETE</status>
    <host>thop76:5476</host>
    <time>17:22:56</time>
    <output>
      <summary>
        <type>neutral</type>
        <xyz unit="nm">-0.1750000 -0.1750000 -5.4250000</xyz>
        <total unit="eV">-3.2112834</total>
        <estat unit="eV">-2.3753255</estat>
        <eindu unit="eV">-0.8359579</eindu>
      </summary>
      <terms_i>
        <F-00-01-11>-3.32999e+00 -3.32481e-01 +4.06829e-02</F-00-01-11>
        <M-00-11-->+1.33689e-01 +4.74490e-01</M-00-11-->
        <E-PP-PU-UU>-2.37533e+00 -7.37812e-01 -2.73212e-18</E-PP-PU-UU>
      </terms_i>
      <terms_o>
        <R-pp-pu-uu>-1.89357e-01 = +2.69583e-08</R-pp-pu-uu>
        <O-pp-pu-uu>+0.00000e+00 = +0.00000e+00</O-pp-pu-uu>
        <C-pp-pu-uu>+1.51186e-01 = +0.00000e+00</C-pp-pu-uu>
      </terms_o>
      <shells>
        <FGC>1874</FGC>
        <FGN>1874</FGN>
        <MGN>54429</MGN>
        <BGN>36</BGN>
        <BGP>52</BGP>
        <QM0>2</QM0>
        <MM1>1872</MM1>
        <MM2>0</MM2>
      </shells>
      <timing>
        <t_total unit="min">5.29</t_total>
        <t_wload unit="min">0.00 2.24 0.88 2.18</t_wload>
      </timing>
    </output>
  </job>
  <job>
    ...
  </job>
  ...
  </jobs>
2.7 Transfer integrals

The electronic transfer integral element $J_{ij}$ entering the Marcus rates in eq. (2.31) is defined as

$$J_{ij} = \langle \phi_i | \hat{H} | \phi_j \rangle,$$

(2.18)  

where $\phi_i$ and $\phi_j$ are diabatic wavefunctions, localized on molecule $i$ and $j$ respectively, participating in the charge transfer, and $\hat{H}$ is the Hamiltonian of the formed dimer. Within the frozen-core approximation, the usual choice for the diabatic wavefunctions $\phi_i$ is the highest occupied molecular orbital (HOMO) in case of hole transport, and the lowest unoccupied molecular orbital (LUMO) in the case of electron transfer, while $\hat{H}$ is an effective single particle Hamiltonian, e.g. Fock or Kohn-Sham operator of the dimer. As such, $J_{ij}$ is a measure of the strength of the electronic coupling of the frontier orbitals of monomers mediated by the dimer interactions.

Intrinsically, the transfer integral is very sensitive to the molecular arrangement, i.e. the distance and the mutual orientation of the molecules participating in charge transport. Since this arrangement can also be significantly influenced by static and/or dynamic disorder [25–29], it is essential to calculate $J_{ij}$ explicitly for each hopping pair within a realistic morphology. Considering that the number of dimers for which eq. (2.18) has to be evaluated is proportional to the number of molecules times their coordination number, computationally efficient and at the same time quantitatively reliable schemes are required.

2.7.1 Projection of monomer orbitals on dimer orbitals (DIPRO)

An approach for the determination of the transfer integral that can be used for any single-particle electronic structure method (Hartree-Fock, DFT, or semiempirical methods) is based on the projection of monomer orbitals on a manifold of explicitly calculated dimer orbitals. This dimer projection (DIPRO) technique including an assessment of computational parameters such as the basis set, exchange-correlation functionals, and convergence criteria is presented in detail in ref. [5]. A brief summary of the concept is given below.

We start from an effective Hamiltonian

$$\hat{H}_{\text{eff}} = \sum_i e_i \hat{a}_i^\dagger \hat{a}_i + \sum_{j \neq i} J_{ij} \hat{a}_i^\dagger \hat{a}_j + \text{c.c.}$$

(2.19)  

where $\hat{a}_i^\dagger$ and $\hat{a}_i$ are the creation and annihilation operators for a charge carrier located at the molecular site $i$. The electron site energy is given by $e_i$, while $J_{ij}$ is the transfer integral between two sites $i$ and $j$. We label their frontier orbitals (HOMO for hole transfer, LUMO for electron transfer) $\phi_i$ and $\phi_j$, respectively. Assuming that the frontier orbitals of a dimer (adiabatic energy surfaces) result exclusively from the interaction of the frontier orbitals of monomers, and consequently expand them in terms of $\phi_i$ and $\phi_j$. The expansion coefficients, $C$, can be determined by solving the secular equation

$$(\hat{H} - ES)\bar{C} = 0$$

(2.20)  

where $\hat{H}$ and $S$ are the Hamiltonian and overlap matrices of the system, respectively. These matrices can be written explicitly as

$$\hat{H} = \begin{pmatrix} e_i & H_{ij} \\ H_{ij}^* & e_j \end{pmatrix} \quad S = \begin{pmatrix} 1 & S_{ij} \\ S_{ij}^* & 1 \end{pmatrix}$$

(2.21)  

with

$$e_i = \langle \phi_i | \hat{H} | \phi_i \rangle \quad H_{ij} = \langle \phi_i | \hat{H} | \phi_j \rangle$$

$$e_j = \langle \phi_j | \hat{H} | \phi_j \rangle \quad S_{ij} = \langle \phi_j | \phi_i \rangle$$

(2.22)  

\[^1\text{we use following notations: } a - \text{number, } \bar{a} - \text{vector, } A - \text{matrix, } \hat{A} - \text{operator}\]
The matrix elements $e_{ij}$, $H_{ij}$, and $S_{ij}$ entering eq. (2.21) can be calculated via projections on the dimer orbitals (eigenfunctions of $\hat{H}$) $\{|\phi_n^D\rangle\}$ by inserting $\hat{1} = \sum_n |\phi_n^D\rangle \langle \phi_n^D|$ twice. We exemplify this explicitly for $H_{ij}$ in the following

$$H_{ij} = \sum_{n,m} \langle \phi_i | \phi_n^D \rangle \langle \phi_n^D | \hat{H} | \phi_m^D \rangle \langle \phi_m^D | \phi_j \rangle. \quad (2.23)$$

The Hamiltonian is diagonal in its eigenfunctions, $\langle \phi_n^D | \hat{H} | \phi_m^D \rangle = E_n \delta_{nm}$. Collecting the projections of the frontier orbitals $|\phi_{i(j)}\rangle$ on the $n$-th dimer state $(\hat{V}_i)_n = \langle \phi_i | \phi_n^D \rangle$ and $(\hat{V}_j)_n = \langle \phi_j | \phi_n^D \rangle$ respectively, into vectors we obtain

$$H_{ij} = \hat{V}_i \hat{E} \hat{V}_j^\dagger. \quad (2.24)$$

What is left to do is determine these projections $\hat{V}_k$. In all practical calculations the molecular orbitals are expanded in basis sets of either plane waves or of localized atomic orbitals $|\varphi_\alpha\rangle$. We will first consider the case that the calculations for the monomers are performed using a counterpoise basis set that is commonly used to deal with the basis set superposition error (BSSE).

The basis set of atom-centered orbitals of a monomer is extended to the one of the dimer by adding the respective atomic orbitals at virtual coordinates of the second monomer. We can then write the respective expansions as

$$|\phi_k\rangle = \sum_\alpha \lambda^{(k)}_\alpha |\varphi_\alpha\rangle \quad \text{and} \quad |\phi_n^D\rangle = \sum_\alpha D^{(n)}_\alpha |\varphi_\alpha\rangle \quad (2.25)$$

where $k = i, j$. The projections can then be determined within this common basis set as

$$(\hat{V}_k)_n = \langle \phi_k | \phi_n^D \rangle = \sum_\alpha \lambda^{(k)}_\alpha \langle \alpha | \sum_\beta D^{(n)}_\beta |\beta\rangle = \bar{\lambda}^\dagger \bar{S} \bar{D} \bar{S}^\dagger \bar{\lambda} \quad (2.26)$$

where $\bar{S}$ is the overlap matrix of the atomic basis functions. This allows us to finally write the elements of the Hamiltonian and overlap matrices in eq. (2.21) as:

$$H_{ij} = \bar{\lambda}^\dagger |\bar{i}\rangle \bar{S} \bar{D} \bar{E} \bar{S}^\dagger |\bar{j}\rangle \quad (2.27)$$

Since the two monomer frontier orbitals that form the basis of this expansion are not orthogonal in general ($\bar{S} \neq 1$), it is necessary to transform eq. (2.20) into a standard eigenvalue problem of the form

$$\mathbf{H}^{\text{eff}} \mathbf{C}^{\text{eff}} \mathbf{e}^{\text{eff}} = E \mathbf{C}^{\text{eff}} \mathbf{e}^{\text{eff}} \quad (2.28)$$

to make it correspond to eq. (2.19). According to Löwdin such a transformation can be achieved by

$$\mathbf{H}^{\text{eff}} = \bar{S}^{-1/2} \mathbf{H} \bar{S}^{-1/2}. \quad (2.29)$$

This then yields an effective Hamiltonian matrix in an orthogonal basis, and its entries can directly be identified with the site energies $\epsilon_i$ and transfer integrals $J_{ij}$:

$$\mathbf{H}^{\text{eff}} = \begin{pmatrix} \epsilon_i & H_{ij}^{\text{eff}} \\ H_{ij}^{\text{eff}} & \epsilon_j \end{pmatrix} = \begin{pmatrix} \epsilon_i & J_{ij} \\ J_{ij}^* & \epsilon_j \end{pmatrix} \quad (2.30)$$

### 2.7.2 DFT-based transfer integrals using DIPRO

The calculation of one electronic coupling element based on DFT using the DIPRO method requires the overlap matrix of atomic orbitals $\bar{S}$, the expansion coefficients for monomer $\bar{\lambda}_{(k)} = \{\lambda^{(k)}_\alpha\}$ and dimer orbitals $\bar{D} \bar{s} = \{D^{(n)}_\alpha\}$, as well as the orbital energies $E_n$ of the dimer are
2.7. TRANSFER INTEGRALS

\[ J_{AB} = H_{AB}^{D} \left( H_{AB}, S_{AB}, \varepsilon_A, \varepsilon_B \right) \]

Figure 2.4: Schematics of the DIPRO method. (a) General workflow of the projection technique. (b) Strategy of the efficient noCP+noSCF implementation, in which the monomer calculations are performed independently from the dimer configurations (noCP), using the edft calculator. The dimer Hamiltonian is subsequently constructed based on an initial guess formed from monomer orbitals and only diagonalized once (noSCF) before the transfer integral is calculated by projection. This second step is performed by the idft calculator.

required as input. In practical situations, performing self-consistent quantum-chemical calculations for each individual monomer and one for the dimer to obtain this input data is extremely demanding. Several simplifications can be made to reduce the computational effort, such as using non-Counterpoise basis sets for the monomers (thereby decoupling the monomer calculations from the dimer run) and performing only a single SCF step in a dimer calculation starting from an initial guess formed from a superposition of monomer orbitals. This “noCP+noSCF” variant of DIPRO is shown in figure 2.4(a) and recommended for production runs. A detailed comparative study of the different variants can be found in [5].

The code currently contains supports evaluation of transfer integrals from quantum-chemical calculations performed with the Gaussian, Turbomole, and NWChem packages. The interfacing procedure consists of three main steps: generation of input files for monomers and dimers, performing the actual quantum-chemical calculations, and calculating the transfer integrals.

Monomer calculations

First, hopping sites and a neighbor list need to be generated from the atomistic topology and trajectory and written to the state.sql file. Then the parallel edft calculator manages the calculation of the monomer properties required for the determination of electronic coupling elements. Specifically, the individual steps it performs are:

1. Creation of a job file containing the list of molecules to be calculated with DFT
2. Running of all jobs in job file

**Running all edft jobs**

ctp_parallel -o options.xml -f state.sql -edft -j run

which includes

- creating the input files for the DFT calculation (using the package specified in options.xml) in the directory
  OR_FILES/package/frame_F/mol_M
  where F is the index of the frame in the trajectory, M is the index of a molecule in this frame,
- executing the DFT run, and
- after completion of this run, parsing the output (number of electrons, basis set, molecular orbital expansion coefficients), and saving it in compressed form to
  OR_FILES/molecules/frame_F/molecule_M.orb

### Calculating the transfer integrals

After the monomer calculations have been completed successfully, the respective runs for dimers from the neighborlist can be performed using the parallel idft calculator, which manages the DFT runs for the hopping pairs and determines the coupling element using DIPRO. Again, several steps are required:

1. **Writing job file for idft**

   ctp_parallel -o options.xml -f state.sql -eidft -j write

2. **Running of all idft jobs**

   ctp_parallel -o options.xml -f state.sql -eidft -j run

which includes

- creating the input files (including the merged guess for a noSCF calculation, if requested) for the DFT calculation (using the package specified in options.xml) in the directory
  OR_FILES/package/frame_F/pair_M_N
  where M and N are the indices of the molecules in this pair,
- executing the DFT run, and
2.8. CHARGE TRANSFER RATE

- after completion of this run, parsing the output (number of electrons, basis set, molecular orbital expansion coefficients and energies, atomic orbital overlap matrix), and saving the pair information in compressed form to

```
OR_FILES/pairs/frame_F/pair_M_N.orb
```

- loading the monomer orbitals from the previously saved *.orb files.
- calculating the coupling elements and write them to the job file

3. Reading the coupling elements from the job file and saving them to the `state.sql` file

```
Saving idft results from job file to state.sql
```

```
ctp_parallel -o options.xml -f state.sql -e idft -j read
```

2.7.3 ZINDO-based transfer integrals using MOO

An approximate method based on Zernier’s Intermediate Neglect of Differential Overlap (ZINDO) has been described in Ref. [7]. This semiempirical method is substantially faster than first-principles approaches, since it avoids the self-consistent calculations on each individual monomer and dimer. This allows to construct the matrix elements of the ZINDO Hamiltonian of the dimer from the weighted overlap of molecular orbitals of the two monomers. Together with the introduction of rigid segments, only a single self-consistent calculation on one isolated conjugated segment is required. All relevant molecular overlaps can then be constructed from the obtained molecular orbitals.

The main advantage of the molecular orbital overlap (MOO) library is fast evaluation of electronic coupling elements. Note that MOO is based on the ZINDO Hamiltonian which has limited applicability. The general advice is to first compare the accuracy of the MOO method to the DFT-based calculations.

MOO can be used both in a standalone mode and as an `izindo` calculator of VOTCA-CTP.

Since MOO constructs the Fock operator of a dimer from the molecular orbitals of monomers by translating and rotating the orbitals of rigid fragments, the optimized geometry of all conjugated segments and the coefficients of the molecular orbitals are required as its input in addition to the state file (`state.sql`) with the neighbor list. Coordinates are stored in `geometry.xyz` files with four columns, first being the atom type and the next three atom coordinates. This is a standard `xyz` format without a header. Note that the atom order in the `geometry.xyz` files can be different from that of the mapping files. The correspondence between the two is established in the `map.xml` file.

### Attention

Izindo requires the specification of orbitals for hole and electron transport in `map.xml`. They are the HOMO and LUMO respectively and can be retrieved from the log file from which the `zindo.orb` file is generated. The number of alpha electrons is the HOMO, the LUMO is HOMO+1

The calculated transfer integrals are immediately saved to the `state.sql` file.

```
Transfer integrals from izindo
```

```
ctp_run -o options.xml -f state.sql -e izindo
```
2.8 Charge transfer rate

Charge transfer rates can be postulated based on intuitive physical considerations, as it is done in the Gaussian disorder models \[25, 30–32\]. Alternatively, charge transfer theories can be used to evaluate rates from quantum chemical calculations \[3, 5, 13, 33–35\]. In spite of being significantly more computationally demanding, the latter approach allows to link the chemical and electronic structure, as well as the morphology, to charge dynamics.

2.8.1 Classical charge transfer rate

The high temperature limit of classical charge transfer theory \[36, 37\] is often used as a trade-off between theoretical rigor and computational complexity. It captures key parameters which influence charge transport while at the same time providing an analytical expression for the rate. Within this limit, the transfer rate for a charge to hop from a site \(i\) to a site \(j\) reads

\[
\omega_{ij} = \frac{2\pi}{\hbar} \frac{|J_{ij}|^2}{\sqrt{\lambda_{ij}^2 k_B T}} \exp \left[ -\frac{(\Delta E_{ij} - \lambda_{ij})^2}{4\lambda_{ij}^2 k_B T} \right].
\]

(2.31)

where \(T\) is the temperature, \(\lambda_{ij} = \lambda_{ij}^{\text{int}} + \lambda_{ij}^{\text{out}}\) is the reorganization energy, which is a sum of intra- and inter-molecular (outersphere) contributions, \(\Delta E_{ij}\) is the site-energy difference, or driving force, and \(J_{ij}\) is the electronic coupling element, or transfer integral.

2.8.2 Semi-classical bimolecular rate

The main assumptions in eq. (2.31) are non-adiabaticity (small electronic coupling and charge transfer between two diabatic, non-interacting states), and harmonic promoting modes, which are treated classically. At ambient conditions, however, the intramolecular promoting mode, which roughly corresponds to C-C bond stretching, has a vibrational energy of \(\hbar \omega \approx 0.2 \text{ eV} \gg k_B T\) and should be treated quantum-mechanically. The outer-sphere (slow) mode has much lower vibrational energy than the intramolecular promoting mode, and therefore can be treated classically. The weak interaction between molecules also implies that each molecule has its own, practically independent, set of quantum mechanical degrees of freedom.

A more general, quantum-classical expression for a bimolecular multi-channel rate is derived in the Supporting Information of ref. \[3\] and has the following form

\[
\omega_{ij} = \frac{2\pi}{\hbar} \frac{|J_{ij}|^2}{\sqrt{\lambda_{ij}^2 k_B T}} \sum_{\nu, \nu' = 0}^{\infty} |\langle \chi_{i0}^c | \chi_{\nu}^c \rangle|^2 |\langle \chi_{j0}^{n'} | \chi_{\nu'}^{n'} \rangle|^2 \exp \left\{ -\frac{[\Delta E_{ij} - \hbar (\nu \omega_i^c + \nu' \omega_j^{n'}) - \lambda_{ij}^{\text{out}}]^2}{4\lambda_{ij}^2 k_B T} \right\}.
\]

(2.32)

If the curvatures of intramolecular PES of charged and neutral states of a molecule are different, that is \(\omega_i^c \neq \omega_i^n\), the corresponding reorganization energies, \(\lambda_i^{cn} = \frac{1}{2} [\omega_i^n (q_i^n - q_i^c)]^2\) and \(\lambda_i^{nc} = \frac{1}{2} [\omega_i^c (q_i^c - q_i^n)]^2\), will also differ. In this case the Franck-Condon (FC) factors for discharging of molecule \(i\) read \[38\]

\[
|\langle \chi_{i0}^c | \chi_{\nu}^n \rangle|^2 = \frac{2}{2^\nu \nu!} \frac{\sqrt{\omega_i^c \omega_i^n}}{\omega_i^c + \omega_i^n} \exp (-s_i) \sum_{k=0}^{\nu} \frac{\nu!}{k! (k/2)!} \left( \frac{2 \omega_i^c}{\omega_i^c + \omega_i^n} \right)^{k/2} \frac{k!}{s_i^k} H_{\nu-k} \left( \frac{s_i}{\sqrt{2s_i^{nc}}} \right)^2,
\]

(2.33)

where \(H_n(x)\) is a Hermite polynomial, \(s_i = 2 \sqrt{\lambda_i^{nc} \lambda_i^{cn}} / (\hbar \omega_i^c + \omega_i^n)\), and \(S_i^{cn} = \lambda_i^{cn} / (\hbar \omega_i^c)\). The FC factors for charging of molecule \(j\) can be obtained by substituting \((s_i, S_i^{cn}, \omega_i^c)\) with \((-s_j, S_j^{nc}, \omega_j^n)\).

In order to evaluate the FC factors, the internal reorganization energy \(\lambda_i^{cn}\) can be computed from the intramolecular PES.
2.9. MASTER EQUATION

2.8.3 Semi-classical rate

One can also use the quantum-classical rate with a common set of vibrational coordinates [14]

\[
\omega_{ij} = \frac{2\pi}{\hbar} \frac{|J_{ij}|^2}{\sqrt{4\pi \lambda_{ij}} \hbar k T} \sum_{N=0}^{\infty} \frac{1}{N!} \left( \frac{\lambda_{ij}^{\text{int}}}{\hbar \omega_{ij}^{\text{int}}} \right)^N \exp \left( -\frac{\lambda_{ij}^{\text{int}}}{\hbar \omega_{ij}^{\text{int}}} \right) \exp \left\{ -\frac{[\Delta E_{ij} - h N \omega_{ij}^{\text{int}} - \lambda_{ij}^{\text{out}}]^2}{4 \lambda_{ij}^{\text{out}} k T} \right\}.
\]

(2.34)

Numerical estimates show that if \( \lambda_{ij}^{\text{int}} \approx \lambda_{ij}^{\text{out}} \) and \(|\Delta E_{ij}| \ll \lambda_{ij}^{\text{out}}\) the rates are similar to those of eq. (2.31). In general, there is no robust method to compute \( \lambda_{ij}^{\text{out}} \) and both reorganization energies are often assumed to be of the same order of magnitude. In this case the second condition also holds, unless there are large differences in electron affinities or ionization potentials of neighboring molecules, e.g. in donor-acceptor blends.

To calculate rates of the type specified in options.xml for all pairs in the neighbor list and to save them into the state.sql file, run the rates calculator. Note that all required ingredients (reorganization energies, transfer integrals, and site energies have to be calculated before).

Calculation of transfer rates

\[\text{kmc}\_\text{run} \quad -o \quad \text{options.xml} \quad -f \quad \text{state.sql} \quad -e \quad \text{rates}\]

2.9 Master equation

Having determined the list of conjugated segments (hopping sites) and charge transfer rates between them, the next task is to solve the master equation which describes the time evolution of the system

\[
\frac{\partial P_\alpha}{\partial t} = \sum_\beta P_\beta \Omega_{\beta\alpha} - \sum_\beta P_\alpha \Omega_{\alpha\beta},
\]

(2.35)

where \( P_\alpha \) is the probability of the system to be in a state \( \alpha \) at time \( t \) and \( \Omega_{\alpha\beta} \) is the transition rate from state \( \alpha \) to state \( \beta \). A state \( \alpha \) is specified by a set of site occupations, \( \{\alpha_i\} \), where \( \alpha_i = 1(0) \) for an occupied (unoccupied) site \( i \), and the matrix \( \Omega \) can be constructed from rates \( \omega_{ij} \).

The solution of eq. (2.35) is be obtained by using kinetic Monte Carlo (KMC) methods. KMC explicitly simulates the dynamics of charge carriers by constructing a Markov chain in state space and can find both stationary and transient solutions of the master equation. The main advantage of KMC is that only states with a direct link to the current state need to be considered at each step. Since these can be constructed solely from current site occupations, extensions to multiple charge carriers (without the mean-field approximation), site-occupation dependent rates (needed for the explicit treatment of Coulomb interactions), and different types of interacting particles and processes, are straightforward. To optimize memory usage and efficiency, a combination of the variable step size method [40] and the first reaction method is implemented.

To obtain the dynamics of charges using KMC, the program kmc_run executes a specific calculator after reading its options (charge carrier type, runtime, number of carriers etc.) from options.xml.

KMC for a single carrier in periodic boundary conditions

\[\text{kmc}\_\text{run} \quad -o \quad \text{options.xml} \quad -f \quad \text{state.sql} \quad -e \quad \text{kmc}\_\text{single}\]
2.9.1 Extrapolation to nondispersive mobilities

Predictions of charge-carrier mobilities in partially disordered semiconductors rely on charge transport simulations in systems which are only several nanometers thick. As a result, simulated charge transport might be dispersive for materials with large energetic disorder [41, 42] and simulated mobilities are system-size dependent. In time-of-flight (TOF) experiments, however, a typical sample thickness is in the micrometer range and transport is often nondispersive. In order to link simulation and experiment, one needs to extract the nondispersive mobility from simulations of small systems, where charge transport is dispersive at room temperature. Such extrapolation is possible if the temperature dependence of the nondispersive mobility is known in a wide temperature range. For example, one can use analytical results derived for one-dimensional models [43–45]. The mobility-temperature dependence can then be parametrized by simulating charge transport at elevated temperatures, for which transport is nondispersive even for small system sizes. This dependence can then be used to extrapolate to the nondispersive mobility at room temperature [4]. For Alq$_3$, the charge carrier mobility of a periodic system of 512 molecules was shown to be more than three orders of magnitude higher than the nondispersive mobility of an infinitely large system [4]. Furthermore, it was shown that the transition between the dispersive and nondispersive transport has a logarithmic dependence on the number of hopping sites $N$. Hence, a brute-force increase of the system size cannot resolve the problem for compounds with large energetic disorder $\sigma$, since $N$ increases exponentially with $\sigma^2$.

2.10 Macroscopic observables

Spatial distributions of charge and current densities can provide a better insight in the microscopic mechanisms of charge transport. If $O$ is an observable which has a value $O_{\alpha}$ in a state $\alpha$, its ensemble average at time $t$ is a sum over all states weighted by the probability $P_{\alpha}$ to be in a state $\alpha$ at time $t$

$$\langle O \rangle = \sum_{\alpha} O_{\alpha} P_{\alpha}. \quad (2.36)$$  

If $O$ does not explicitly depend on time, the time evolution of $\langle O \rangle$ can be calculated as

$$\frac{d \langle O \rangle}{dt} = \sum_{\alpha,\beta} [P_{\beta} \Omega_{\beta\alpha} - P_{\alpha} \Omega_{\alpha\beta}] O_{\alpha} = \sum_{\alpha,\beta} P_{\beta} \Omega_{\beta\alpha} \left[ O_{\alpha} - O_{\beta} \right]. \quad (2.37)$$

If averages are obtained from KMC trajectories, $P_{\alpha} = s_{\alpha}/s$, where $s_{\alpha}$ is the number of Markov chains ending in the state $\alpha$ after time $t$, and $s$ is the total number of chains. Alternatively, one can calculate time averages by analyzing a single Markov chain. If the total occupation time of the state $\alpha$ is $\tau_{\alpha}$, then

$$\overline{O} = \frac{1}{\tau} \sum_{\alpha} O_{\alpha} \tau_{\alpha}, \quad (2.38)$$

where $\tau = \sum_{\alpha} \tau_{\alpha}$ is the total time used for time averaging.

For ergodic systems and sufficient sampling times, ensemble and time averages should give identical results. In many cases, the averaging procedure reflects a specific experimental technique. For example, an ensemble average over several KMC trajectories with different starting conditions corresponds to averaging over injected charge carriers in a time-of-flight experiment. In what follows, we focus on the single charge carrier (low concentration of charges) case.
2.10.1 Charge density

For a specific type of particles, the microscopic charge density of a site $i$ is proportional to the occupation probability of the site, $p_i$,

$$p_i = \frac{e p_i}{V_i},$$

(2.39)

where, for an irregular lattice, the effective volume $V_i$ can be obtained from a Voronoi tessellation of space. For reasonably uniform lattices (uniform site densities) this volume is almost independent of the site and a constant volume per site, $V_i = V/N$, can be assumed. In the macroscopic limit, the charge density can be calculated using a smoothing kernel function, i.e. a distance-weighted average over multiple sites. Site occupations $p_i$ can be obtained from eq. (2.36) or eq. (2.38) by using the occupation of site $i$ in state $\alpha$ as an observable.

If the system is in thermodynamic equilibrium, that is without sources or sinks and without circular currents (and therefore no net flux) a condition known as detailed balance, holds

$$p_j \omega_{ji} = p_i \omega_{ij},$$

(2.40)

equ:detailed_balance

It can be used to test whether the system is ergodic or not by correlating $\log p_i$ and the site energy $E_i$. Indeed, if $\lambda_{ij} = \lambda_{ji}$ the ratios of forward and backward rates are determined solely by the energetic disorder, $\omega_{ji}/\omega_{ij} = \exp(-\Delta E_{ij}/k_B T)$ (see eq. (2.31)).

2.10.2 Current

If the position of the charge, $\vec{r}$, is an observable, the time evolution of its average $\langle \vec{r} \rangle$ is the total current in the system

$$\vec{J} = e \langle \vec{v} \rangle = e \frac{d}{dt} \langle \vec{r} \rangle = e \sum_{i,j} p_j \omega_{ji} (\vec{r}_i - \vec{r}_j).$$

(2.41)

equ:current_def

Symmetrizing this expression we obtain

$$\vec{J} = \frac{1}{2} e \sum_{i,j} (p_j \omega_{ji} - p_i \omega_{ij}) \vec{r}_{ij},$$

(2.42)

equ:current

where $\vec{r}_{ij} = \vec{r}_i - \vec{r}_j$. Symmetrization ensures equal flux splitting between neighboring sites and absence of local average fluxes in equilibrium. It allows to define a local current through site $i$ as

$$\vec{J}_i = \frac{1}{2} e \sum_{j} (p_j \omega_{ji} - p_i \omega_{ij}) \vec{r}_{ij}.$$

(2.43)

equ:site_current

A large value of the local current indicates that the site contributes considerably to the total current. A collection of such sites thus represents most favorable charge pathways [46].

2.10.3 Mobility and diffusion constant

For a single particle, e.g. a charge or an exciton, a zero-field mobility can be determined by studying particle diffusion in the absence of external fields. Using the particle displacement squared, $\Delta \vec{r}_i^2$, as an observable we obtain

$$2 d D_{\gamma \delta} = \frac{d}{dt} \langle \Delta \vec{r}_{i,\gamma} \Delta \vec{r}_{i,\delta} \rangle = \sum_{i,j} p_j \omega_{ji} (\Delta \vec{r}_{i,\gamma} \Delta \vec{r}_{i,\delta} - \Delta \vec{r}_{j,\gamma} \Delta \vec{r}_{j,\delta}) = \sum_{i,j} p_j \omega_{ji} (\vec{r}_{i,\gamma} \vec{r}_{i,\delta} - \vec{r}_{j,\gamma} \vec{r}_{j,\delta}).$$

(2.44)

equ:diffusion

Here $\vec{r}_i$ is the coordinate of the site $i$, $D_{\gamma \delta}$ is the diffusion tensor, $\gamma, \delta = x, y, z$, and $d = 3$ is the system dimension. Using the Einstein relation,

$$D_{\gamma \delta} = k_B T \rho_{\gamma \delta},$$

(2.45)
one can, in principle, obtain the zero-field mobility tensor $\mu_{\gamma \delta}$. Eq. (2.44), however, does not take
into account the use of periodic boundary conditions when simulating charge dynamics. In this
case, the simulated occupation probabilities can be compared to the solution of the Smoluchowski
equation with periodic boundary conditions (see the supporting information for details).
Alternatively, one can directly analyze time-evolution of the KMC trajectory and obtain the dif-
fusion tensor from a linear fit to the mean square displacement, $\overline{\Delta r_{ij}} = 2dD_{ij}t$.
The charge carrier mobility tensor, $\mu$, for any value of the external field can be determined either
from the average charge velocity defined in eq. (2.41)
\[
\langle \vec{v} \rangle = \sum_{i,j} p_{ij} \omega_{ji}(\vec{r}_i - \vec{r}_j) = \mu \vec{F},
\]
or directly from the KMC trajectory. In the latter case the velocity is calculated from the un-
wrapped (if periodic boundary conditions are used) charge displacement vector divided by the
total simulation time. Projecting this velocity on the direction of the field $\vec{F}$ yields the charge car-
rier mobility in this particular direction. In order to improve statistics, mobilities can be averaged
over several KMC trajectories and MD snapshots.

### 2.10.4 Spatial correlations of energetic disorder

Long-range, e.g. electrostatic and polarization, interactions often result in spatially correlated
disorder [47], which affects the onset of the mobility-field (Poole-Frenkel) dependence [43, 48, 49].
To quantify the degree of correlation, one can calculate the spatial correlation function of $E_i$ and
$E_j$ at a distance $r_{ij}$
\[
C(r_{ij}) = \frac{\langle (E_i - \langle E_i \rangle)(E_j - \langle E_j \rangle) \rangle}{\langle (E_i - \langle E_i \rangle)^2 \rangle},
\]
where $\langle E \rangle$ is the average site energy. $C(r_{ij})$ is zero if $E_i$ and $E_j$ are uncorrelated and 1 if they are
fully correlated. For a system of randomly oriented point dipoles, the correlation function decays
as $1/r$ at large distances [50].
For systems with spatial correlations, variations in site energy differences, $\Delta E_{ij}$, of pairs of
molecules from the neighbor list are smaller than variations in site energies, $E_i$, of all individual
molecules. Since only neighbor list pairs affect transport, the distribution of $\Delta E_{ij}$ rather than
that of individual site energies, $E_i$, should be used to characterize energetic disorder.
Note that the eanalyze calculator takes into account all contributions to the site energies

```
ctp_run -o options.xml -f state.sql -e eanalyze
```

### 2.10.5 DFT-based transfer integrals using DIPRO

The calculation of one electronic coupling element based on DFT using the DIPRO method re-
quires the overlap matrix of atomic orbitals $S$, the expansion coefficients for monomer $\lambda_{(k)} =
\{\lambda_{n}^{(k)}\}$ and dimer orbitals $\tilde{D}_{(n)} = \{D_{n}^{(n)}\}$, as well as the orbital energies $E_n$ of the dimer
are required as input. In practical situations, performing self-consistent quantum-chemical calcula-
tions for each individual monomer and one for the dimer to obtain this input data is extremely
demanding. Several simplifications can be made to reduce the computational effort, such as
using non-Counterpoise basis sets for the monomers (thereby decoupling the monomer calcula-
tions from the dimer run) and performing only a single SCF step in a dimer calculation starting
from an initial guess formed from a superposition of monomer orbitals. This "noCP+noSCF" variant of DIPRO is shown in figure 2.4(a) and recommended for production runs. A detailed
comparative study of the different variants can be found in [5].
The code currently contains supports evaluation of transfer integrals from quantum-chemical calculations performed with the Gaussian, Turbomole, and NWChem packages. The interfacing procedure consists of three main steps: generation of input files for monomers and dimers, performing the actual quantum-chemical calculations, and calculating the transfer integrals.

### Monomer calculations

First, hopping sites and a neighbor list need to be generated from the atomistic topology and trajectory and written to the `state.sql` file. Then the parallel edft calculator manages the calculation of the monomer properties required for the determination of electronic coupling elements. Specifically, the individual steps it performs are:

1. Creation of a job file containing the list of molecules to be calculated with DFT

   ```
   Writing job file for edft
   ctp_parallel -o options.xml -f state.sql -e edft -j write
   ```

2. Running of all jobs in job file

   ```
   Running all edft jobs
   ctp_parallel -o options.xml -f state.sql -e edft -j run
   ```

   which includes

   - creating the input files for the DFT calculation (using the package specified in `options.xml`) in the directory
     
     \[ \text{OR_FILES/package/frame}_F/mol_M \]
     
     where \( F \) is the index of the frame in the trajectory, \( M \) is the index of a molecule in this frame,
   - executing the DFT run, and
   - after completion of this run, parsing the output (number of electrons, basis set, molecular orbital expansion coefficients), and saving it in compressed form to
     
     \[ \text{OR_FILES/molecules/frame}_F/molecule}_M.orb \]

### Calculating the transfer integrals

After the monomer calculations have been completed successfully, the respective runs for dimers from the neighborlist can be performed using the parallel idft calculator, which manages the DFT runs for the hopping pairs and determines the coupling element using DIPRO. Again, several steps are required:

1. Creation of a job file containing the list of pairs to be calculated with DFT

   ```
   Writing job file for idft
   ctp_parallel -o options.xml -f state.sql -e idft -j write
   ```

2. Running of all jobs in job file
Running all `idft` jobs

```
ctp_parallel -o options.xml -f state.sql -e idft -j run
```

which includes

- creating the input files (including the merged guess for a noSCF calculation, if requested) for the DFT calculation (using the package specified in `options.xml`) in the directory
  
  `OR_FILES/package/frame_F/pair_M_N`

  where M and N are the indices of the molecules in this pair,

- executing the DFT run, and

- after completion of this run, parsing the output (number of electrons, basis set, molecular orbital expansion coefficients and energies, atomic orbital overlap matrix), and saving the pair information in compressed form to
  
  `OR_FILES/pairs/frame_F/pair_M_N.orb`

- loading the monomer orbitals from the previously saved `*.orb` files.

- calculating the coupling elements and write them to the job file

3. Reading the coupling elements from the job file and saving them to the `state.sql` file

```
Saving `idft` results from job file to `state.sql`
```

```
ctp_parallel -o options.xml -f state.sql -e idft -j read
```
Chapter 3

Input and output files

3.1 Atomistic topology

If you are using GROMACS for generating atomistic configurations, it is possible to directly use
the topology file provided by GROMACS (topology.tpr). In this case the GROMACS residue and
atom names should be used to specify the coarse-grained topology and conjugated segments.
A custom topology can also be defined using an XML file. Moreover, it is possible to partially
overwrite the information provided in, for example, GROMACS topology file. We will illustrate
how to create a custom topology file using DCV2T. The structure of DCV2T, together with atom
type definitions, is shown in fig. 3.1. DCV2T has two thiophene (THI) and two dicyanovinyl
(NIT) residues. The pdb file which contains residue types, residue numbering, atom names,
atom types, and atom coordinates is shown in listing 3.1.

Figure 3.1: (a) DCV2T with atoms labelled according to residue_number:residue_name:atom_name.
There are four residues and two residue types: thiophene (THI) and dicyanovinyl (NIT). The corresponding
pdb file is shown in listing 3.1. Atom numbering is used to split conjugated segments on rigid fragments
and to link atomistic ((b) from GROMACS topology) and quantum descriptions (c).
<table>
<thead>
<tr>
<th>N</th>
<th>Atom</th>
<th>Residue</th>
<th>Chain</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>Temperature</th>
<th>Atomic Species</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>N1</td>
<td>NIT</td>
<td></td>
<td>2.388</td>
<td>8.533</td>
<td>11.066</td>
<td>1.00</td>
<td>N</td>
</tr>
<tr>
<td>2</td>
<td>CN1</td>
<td>NIT</td>
<td></td>
<td>1.984</td>
<td>9.553</td>
<td>10.718</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>3</td>
<td>N2</td>
<td>NIT</td>
<td></td>
<td>-1.138</td>
<td>10.872</td>
<td>10.087</td>
<td>1.00</td>
<td>N</td>
</tr>
<tr>
<td>4</td>
<td>CN2</td>
<td>NIT</td>
<td></td>
<td>0.003</td>
<td>10.871</td>
<td>11.213</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>5</td>
<td>CC1</td>
<td>NIT</td>
<td></td>
<td>1.441</td>
<td>10.824</td>
<td>10.327</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>6</td>
<td>Cl</td>
<td>NIT</td>
<td></td>
<td>2.193</td>
<td>11.939</td>
<td>10.071</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>7</td>
<td>HN1</td>
<td>NIT</td>
<td></td>
<td>1.715</td>
<td>12.710</td>
<td>9.872</td>
<td>1.00</td>
<td>H</td>
</tr>
<tr>
<td>8</td>
<td>S1</td>
<td>THI</td>
<td></td>
<td>4.758</td>
<td>10.743</td>
<td>10.130</td>
<td>1.00</td>
<td>S</td>
</tr>
<tr>
<td>9</td>
<td>CA1</td>
<td>THI</td>
<td></td>
<td>3.613</td>
<td>12.024</td>
<td>9.948</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>10</td>
<td>CA2</td>
<td>THI</td>
<td></td>
<td>6.099</td>
<td>11.836</td>
<td>9.997</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>11</td>
<td>CB1</td>
<td>THI</td>
<td></td>
<td>4.251</td>
<td>13.243</td>
<td>9.782</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>12</td>
<td>CB2</td>
<td>THI</td>
<td></td>
<td>5.658</td>
<td>13.131</td>
<td>9.818</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>13</td>
<td>HC1</td>
<td>THI</td>
<td></td>
<td>3.800</td>
<td>14.047</td>
<td>9.660</td>
<td>1.00</td>
<td>H</td>
</tr>
<tr>
<td>14</td>
<td>HC2</td>
<td>THI</td>
<td></td>
<td>6.230</td>
<td>13.860</td>
<td>9.731</td>
<td>1.00</td>
<td>H</td>
</tr>
<tr>
<td>15</td>
<td>S1</td>
<td>THI</td>
<td></td>
<td>8.803</td>
<td>12.414</td>
<td>9.882</td>
<td>1.00</td>
<td>S</td>
</tr>
<tr>
<td>16</td>
<td>CA1</td>
<td>THI</td>
<td></td>
<td>7.456</td>
<td>11.347</td>
<td>10.094</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>17</td>
<td>CA2</td>
<td>THI</td>
<td></td>
<td>9.940</td>
<td>11.122</td>
<td>10.152</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>18</td>
<td>CB1</td>
<td>THI</td>
<td></td>
<td>7.873</td>
<td>10.048</td>
<td>10.355</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>19</td>
<td>CB2</td>
<td>THI</td>
<td></td>
<td>9.267</td>
<td>9.926</td>
<td>10.399</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>20</td>
<td>HC1</td>
<td>THI</td>
<td></td>
<td>7.288</td>
<td>9.335</td>
<td>10.487</td>
<td>1.00</td>
<td>H</td>
</tr>
<tr>
<td>21</td>
<td>HC2</td>
<td>THI</td>
<td></td>
<td>9.704</td>
<td>9.123</td>
<td>10.576</td>
<td>1.00</td>
<td>H</td>
</tr>
<tr>
<td>22</td>
<td>N1</td>
<td>NIT</td>
<td></td>
<td>11.235</td>
<td>14.572</td>
<td>9.094</td>
<td>1.00</td>
<td>N</td>
</tr>
<tr>
<td>23</td>
<td>CN1</td>
<td>NIT</td>
<td></td>
<td>11.665</td>
<td>13.566</td>
<td>9.441</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>24</td>
<td>N2</td>
<td>NIT</td>
<td></td>
<td>14.733</td>
<td>12.005</td>
<td>10.009</td>
<td>1.00</td>
<td>N</td>
</tr>
<tr>
<td>25</td>
<td>CN2</td>
<td>NIT</td>
<td></td>
<td>13.590</td>
<td>12.149</td>
<td>9.933</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>26</td>
<td>CC1</td>
<td>NIT</td>
<td></td>
<td>12.156</td>
<td>12.282</td>
<td>9.861</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>27</td>
<td>Cl</td>
<td>NIT</td>
<td></td>
<td>11.363</td>
<td>11.220</td>
<td>10.154</td>
<td>1.00</td>
<td>C</td>
</tr>
<tr>
<td>28</td>
<td>HN1</td>
<td>NIT</td>
<td></td>
<td>11.813</td>
<td>10.440</td>
<td>10.389</td>
<td>1.00</td>
<td>H</td>
</tr>
</tbody>
</table>

Listing 3.1: pdb file of DCV2T.
3.2 Mapping file

The mapping file (referred here as map.xml) is used by the program pdb2map to convert an atomistic trajectory to a trajectory with conjugated segments and rigid fragments. This trajectory is stored in a state file and contains positions, names, types of atoms belonging to rigid fragments.

The description of the mapping options is given in Table 3.1. An example of map.xml for a DCV2T molecule is shown in listing 3.2.

The file map.xml contains the whole electrostatic information about the molecules as well as the structural information. The tool pdb2map creates a map.xml from a pdb file and is a good starting point for further refinement.

### Table 3.1: Description of the XML mapping file (map.xml).

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>topology</td>
<td>This file is used to convert an atomistic trajectory to conjugated segments.</td>
</tr>
<tr>
<td>molecule</td>
<td>Contains the whole electrostatic information about the molecule.</td>
</tr>
<tr>
<td>&lt;name&gt;DCV2T-NC1&lt;/name&gt;</td>
<td>Name of the conjugated molecule.</td>
</tr>
<tr>
<td>&lt;name&gt;protein&lt;/name&gt;</td>
<td>Name of the protein.</td>
</tr>
<tr>
<td>&lt;orbital&gt;U_cC_nN_h&lt;/orbital&gt;</td>
<td>Number of the orbital.</td>
</tr>
<tr>
<td>&lt;multipoles_m&gt;U_cC_nN_h&lt;/multipoles_m&gt;</td>
<td>Multipole file for neutral state.</td>
</tr>
<tr>
<td>&lt;multipoles_h&gt;U_cC_nN_h&lt;/multipoles_h&gt;</td>
<td>Multipole file for hole state.</td>
</tr>
<tr>
<td>&lt;map2md&gt;0&lt;/map2md&gt;</td>
<td>Specifies if planar QM coordinates (map2md=0) or MD coordinates (map2md=1) of atoms are used for distribution of partial charges. For MD coordinates the order and numbering in &lt;mdatoms&gt; and &lt;mpoles&gt; must be identical.</td>
</tr>
<tr>
<td>&lt;site_energy&gt;0.1&lt;/site_energy&gt;</td>
<td>Site energy.</td>
</tr>
<tr>
<td>&lt;reorg_discharge&gt;0.1&lt;/reorg_discharge&gt;</td>
<td>Reorganization energy.</td>
</tr>
<tr>
<td>&lt;reorg_charge&gt;0.1&lt;/reorg_charge&gt;</td>
<td>Reorganization charge.</td>
</tr>
<tr>
<td>&lt;MDQMMapping&gt;1&lt;/MDQMMapping&gt;</td>
<td>MD QM Mapping.</td>
</tr>
<tr>
<td>&lt;fragment&gt;</td>
<td>Three atoms: define a cartesian local frame, two atoms: fragment is assumed to be rotationally invariant around the axis, one atom: fragment is assumed isotropic.</td>
</tr>
<tr>
<td>&lt;weights&gt;</td>
<td>Optional flag: says if a site is virtual or not, (virtual=1, real=0).</td>
</tr>
</tbody>
</table>

### Listing 3.2: Example of map.xml for DCV2T. Each rigid fragment (coarse-grained bead) is defined by a list of atoms. Atom numbers, names, and residue names should correspond to those used in GROMACS topology (see the corresponding listing 3.1 of the pdb file).
### 3.3 Conjugated segments

The file describing hopping sites, or conjugated segments, is used by practically all programs and calculators. It links the coarse-grained trajectory (positions and orientations of rigid fragments) and quantum-mechanical descriptions of all conjugated segments. The description of this XML file (segments.xml) is given in table 3.2. An example for DCV2T is shown in listing 3.3.

<table>
<thead>
<tr>
<th>Table 3.2: Description of conjugated segments (segments.xml).</th>
</tr>
</thead>
</table>

```
<virtual_mps> 0 0 0 0 0 0 0 0 </virtual_mps>
</fragment>

<fragment>
  <name>THI</name>
  <qmats> 7:8 8:6 6:9 9:3 </qmats>
  <mpoles> 7:8 8:6 6:9 9:3 </mpoles>
  <weights> 32 12 12 12 12 12 12 12 1 </weights>
  <localframe> 7 8 6 </localframe>
</fragment>

<fragment>
  <name>THI</name>
  <qmats> 3:4 4:3 3:4 3:4 </qmats>
  <weights> 32 12 12 12 12 12 12 1 1 </weights>
  <localframe> 3 4 2 </localframe>
</fragment>

<fragment>
  <name>NI2</name>
  <weights> 14 12 14 12 12 12 12 12 1 1 </weights>
  <localframe> 22 21 18 </localframe>
</fragment>
```

Listing 3.3: XML file describing conjugated segments. Note that the mapping and weights for each segment are separated by a colon.
3.4 Molecular orbitals

If the semi-empirical method is used to calculate electronic coupling elements, molecular orbitals of all molecules must be supplied. They can be generated using Gaussian program. The Gaussian input file for DCV2T is shown in listing 3.4. Provided with this input, Gaussian will generate fort.7 file which contains the molecular orbitals of a DCV2T. This file can be renamed to DCV2T.orb. Note that the order of the atoms in the input file and the order of coefficients should always match. Therefore, the coordinate part of the input file must be supplied together with the orbitals. We will assume the coordinates, in the format atom_type: x y z, is saved to the DCV2T.xyz file.

Attention
Izindo requires the specification of orbitals for hole and electron transport in map.xml. They are the HOMO and LUMO respectively and can be retrieved from the log file from which the DCV2T.orb file is generated. The number of alpha electrons is the HOMO, the LUMO is HOMO+1.

Listing 3.4: Gaussian input file get_orbitals.com used for generating molecular orbitals. The first line contains the name of the check file, the second the requested RAM. int=zindos requests the method ZINDO, punch=mo states that the molecular orbitals ought to be written to the fort.7 file, nosymm forbids use of symmetry and is necessary to ensure correct position of orbitals with respect to the provided coordinates. The two integer numbers correspond to the charge and multiplicity of the system: 01 corresponds to a neutral system with a multiplicity of one. They are followed by the types and coordinates of all atoms in the molecule.

```plaintext
%chk=DCV2T.chk
%mem=100Mb
#p int=zindos punch=mo nosymm

DCV2T molecular orbitals

0 1
S  -1.44650  2.12185  0.00135
C  -2.43098  0.58936 -0.00048
C  -1.59065 -0.51859 -0.00146
C  -0.21222 -0.22233 -0.00095
C   0.07761  1.13376  0.00040
S   2.87651  0.79316  0.00148
C   3.86099  2.32565  0.00235
C   3.02066  3.43359  0.00231
C   1.64223  3.13733  0.00162
C   1.35240  1.78125  0.00114
C  -3.85350  0.52245  0.00081
C  -4.79569  1.52479  0.00008
C  -6.18500  1.18622  0.00017
C  -4.47544  2.91565  0.00081
C   5.28350  2.39256  0.00296
C   6.22569  1.39020  0.00327
C   7.61500  1.72876  0.00432
C   5.90542  0.00064  0.00333
N  -7.32389  0.89743 -0.00195
N  -4.21872  4.06274  0.00142
N  -8.75389  2.01754  0.00510
N   5.64864 -1.14772  0.00361
H  -1.98064 -1.52966 -0.00256
H   0.55785 -0.98374 -0.00169
```
3.5 Monomer calculations for DFT transfer integrals

Listing 3.5: Example package.xml file for the Gaussian package required in the options of the edft calculator for the monomer calculations as preparation for the determination of transfer integrals using DIPRO.

```xml
<package>
  <name>gaussian</name>
  <executable>g09</executable>
  <checkpoint></checkpoint>
  <scratch></scratch>
  <charge>0</charge>
  <spin>1</spin>
  <options># pop=minimal pbepbe/6-311g** scf=tight punch=no nosymm test</options>
  <memory>1Gb</memory>
  <threads>2</threads>
  <cleanup></cleanup>
</package>
```

Listing 3.6: Example package.xml file for the Turbomole package required in the options of the edft calculator for the monomer calculations as preparation for the determination of transfer integrals using DIPRO.

```xml
<package>
  <name>turbomole</name>
  <executable>ridft</executable>
  <scratch>/tmp</scratch>
  <options>
    TITLE a coord
    no
    b all def-TZVP
    eht
    y
    0
    y
dft
    on
    func
grid
    pbe
    pbe
    grid
    d
    m3
    #
    ri
    on
    m 300
    #
    scf
    conv
    7
    iter
    200
  </options>
</package>
```
3.6 Pair calculations for DFT transfer integrals

Listing 3.7: Example package.xml file for the NWChem package required in the options of the edft calculator for the monomer calculations as preparation for the determination of transfer integrals using DIPRO.

Listing 3.8: Example package.xml file for the Gaussian package required in the options of the idft calculator for the pair calculations and the determination of transfer integrals using DIPRO.

Listing 3.9: Example package.xml file for the Turbomole package required in the options of the idft calculator for the pair calculations and the determination of transfer integrals using DIPRO.
Listing 3.10: Example package.xml file for the NWChem package required in the options of the idft calculator for the pair calculations and the determination of transfer integrals using DIPRO.
3.7 DFT transfer integrals

Listing 3.11: Example TI.xml file created as the output of a DIPRO calculation. Due to slightly different implementations, the orbitals indices refer to monomer indices in a Gaussian run but to indices in the merged dimer guess in a Turbomole run.
3.8 State file

All data structures are saved to the state.sql file in sqlite3 format, see http://www.sqlite.org/.

They are available in form of tables in the state.sql file as can be seen by the command

```
sqlite3 state.sql " .tables "
```

An example of such a table are molecules. The full table can be displayed using the command

```
sqlite3 state.sql " SELECT * FROM molecules "
```

The meaning of all the entries in the table can be displayed by a command like

```
sqlite3 state.sql " .SCHEMA molecules "
```

The first and second entry are integers for internal and regular id of the molecule and the third entry is the name. A single field from the table like the name of the molecule can be displayed by a command like

```
sqlite3 state.sql " SELECT name FROM molecules "
```

Besides molecules, the following tables are stored in the state.sql:

- **conjsge_properties**
  - Conjugated segments are stored with id, name and x,y,z coordinates of the center of mass in nm.
  - **conjsge**: Reorganization energies for charging or discharging a conjugated segment are stored together with the coulomb energy and any other user defined energy contribution (in eV) and occupation probabilities.
  - **pairs**: The pairs from the neighborlist are stored with the pair id, the id of the first and second segment, the rate from the first to the second , the rate from the second to the first (both in s\(^{-1}\)) and the x,y,z coordinates in nm of the distance between the first and the second segment.
  - **pairintegrals**: Transfer integrals for all pairs are stored in the following way: The pair id , the number for counting possible different electronic overlaps (e.g if only the frontier orbitals are taken into account this is always zero, while an effective value is stored in addition to the different overlaps of e.g. HOMO-1 and HOMO-1 if more frontier orbitals are taken into account) and the integral in eV.
  - **pairproperties**: The outer sphere reorganization energy of all pairs is stored by an id, the pair id, a string like lambda_outer and the energy in eV.
  - **conjsge**: Conjugated segments are saved in the following way: The id, the name, the type, the molecule id, the time frame, the x,y,z coordinates in nm and the occupation probability.
  - **conjsge_properties**: Properties of the conjugated segments like reorganization energies for charging or discharging a charge unit or the coulomb contribution to the site energy are stored by: id, conjugated segment id, a string like lambda_intra_charging, lambda_intra_discharging or energy_coulomb and a corresponding value in eV.
  - The tables rigidfrag_properties, rigidfrags and frames offer information about rigid fragments and time frames including periodic boundary conditions.
  - The data in the state.sql file can also be modified by the user. Here is an example how to modify the transfer integral between the conjugated segments number one and two assuming that they are in the neighborlist. Their pair id can be found by the command

```sql
sqlite3 state.sql " UPDATE pairintegrals SET integral = 0.5 WHERE pair_id = 1 AND pair_id2 = 2 "
```
3.8. STATE FILE

381  pair_ID='sqlite3 state.sql "SELECT _id FROM pairs WHERE conjseg1=1 AND conjseg2=2"'
382  sqlite3 state.sql "DELETE FROM pair_integrals WHERE pair=$pair_ID"
383
384  Finally the new transfer integral \( J \) can be written to the state.sql file by the command
385  sqlite3 state.sql "INSERT INTO pair_integrals (pair,num,J) VALUES ($pair_ID,0,$J)"
386
387  Here the \( \text{num}=0 \) indicates that only the effective transfer integrals is written to the file, while other
388  values of \( \text{num} \) would correspond to overlap between other orbitals than the frontier orbitals.
389
390  In a similar way the coulomb contribution to the site energy of the first conjugated segment can
391  be overwritten by first getting its id
392  c_ID='sqlite3 state.sql "SELECT _id from conjseg_properties where conjseg=1 AND
393      key ="energy_coulomb""
394  Then deleting the old value
395  sqlite3 state.sql "DELETE FROM from conjseg_properties WHERE _id=$c_ID"
396  Then the new coulomb energy \( E \) can be written to this id
397  sqlite3 state.sql "INSERT INTO conjseg_properties (_id,conjseg,key,value)
398      VALUES ($c_ID,1,"energy_coulomb",$E)"
399  Finally the resulting coulomb contribution to all conjugated segments can be displayed by
400  sqlite3 state.sql "SELECT * from conjseg_properties WHERE key="energy_coulomb""
Chapter 4

Reference

4.1 Programs

Programs execute specific tasks (calculators).

4.1.1 ctp_map

Generates QM|MD topology

- h [ --help ] display this help and exit
- v [ --verbose ] be loud and noisy
- t [ --topology ] arg topology
- c [ --coordinates ] arg coordinates or trajectory
- s [ --segments ] arg definition of segments and fragments
- f [ --file ] arg state file
--man output man-formatted manual pages
--tex output tex-formatted manual pages

4.1.2 ctp_dump

Extracts information from the state file

- h [ --help ] display this help and exit
- v [ --verbose ] be loud and noisy
- o [ --options ] arg calculator options
- f [ --file ] arg sqlight state file, *.sql
- l [ --first-frame ] arg (=1) start from this frame
- n [ --nframes ] arg (=1) number of frames to process
- t [ --nthreads ] arg (=1) number of threads to create
- s [ --save ] arg (=1) whether or not to save changes to state file
- e [ --extract ] arg List of extractors separated by ’,’ or ’’
- l [ --list ] Lists all available extractors
- d [ --description ] arg Short description of an extractor
--man output man-formatted manual pages
--tex output tex-formatted manual pages

4.1.3 ctp_tools

Runs charge transport tools

- h [ --help ] display this help and exit
- v [ --verbose ] be loud and noisy
4.1.4 ctp_run

Runs charge transport calculators

```
-t [ --nthreads ] arg (=1) number of threads to create
-o [ --options ] arg calculator options
--man output man-formatted manual pages
--tex output tex-formatted manual pages
-e [ --execute ] List of tools separated by ',', or ''
-l [ --list ] Lists all available tools
-d [ --description ] arg Short description of a tool
```

4.1.5 ctp_parallel

Runs job-based heavy-duty calculators

```
-h [ --help ] display this help and exit
-v [ --verbose ] be loud and noisy
-o [ --options ] arg calculator options
-f [ --file ] arg sqlite state file, *.sql
-i [ --first-frame ] arg (=1) start from this frame
-n [ --nframes ] arg (=1) number of frames to process
-t [ --nthreads ] arg (=1) number of threads to create
-s [ --save ] arg (=1) whether or not to save changes to state file
-e [ --execute ] arg List of calculators separated by ',', or ''
-l [ --list ] Lists all available calculators
-d [ --description ] arg Short description of a calculator
--man output man-formatted manual pages
--tex output tex-formatted manual pages
```

4.1.6 moo_overlap

```
-h [ --help ] display this help and exit
-v [ --verbose ] be loud and noisy
--man output man-formatted manual pages
--tex output tex-formatted manual pages
--conjseg arg xml file describing two conjugated segments
--pos1 arg position and orientation of molecule 1
```
4.2 Calculators

Calculator is a piece of code which computes specific system properties, such as site energies, transfer integrals, etc. ctp_run, kmc_run are wrapper programs which executes such calculators. The generic syntax is

ctp_run -e "calc1, calc2, ..." -o options.xml

File options.xml lists all options needed to run a specific calculator. The format of this file is explained in listing 4.1. A complete list of calculators is given in the calculators reference section.

Listing 4.1: A part of the options.xml file with options for the calculator_name{1,2} calculators.

A list of all calculators and their short descriptions can be obtain using

ctp_run --list

A detailed description of all options of a specific calculator(s) is available via

ctp_run --desc calc1,calc2,...

4.2.1 coupling

Electronic couplings from log and orbital files (GAUSSAIN, TURBOMOLE, NWChem)

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>package</td>
<td>coupling.out.xml</td>
<td></td>
<td>First-principles package</td>
</tr>
<tr>
<td>output</td>
<td>coupling.out.xml</td>
<td></td>
<td>Output file</td>
</tr>
<tr>
<td>degeneracy</td>
<td>0</td>
<td>eV</td>
<td>Criterium for the degeneracy of two levels</td>
</tr>
<tr>
<td>moleculeA</td>
<td>A.log</td>
<td></td>
<td>Log file of molecule A</td>
</tr>
<tr>
<td>log</td>
<td>A.orb</td>
<td></td>
<td>Orbitals file</td>
</tr>
<tr>
<td>levels</td>
<td>3</td>
<td></td>
<td>Output HOMO, ..., HOMO-levels; LUMO, ..., LUMO+levels</td>
</tr>
<tr>
<td>trim</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>moleculeB</td>
<td>B.log</td>
<td></td>
<td>Log file of molecule B</td>
</tr>
<tr>
<td>log</td>
<td>B.orb</td>
<td></td>
<td>Orbitals file</td>
</tr>
<tr>
<td>levels</td>
<td>3</td>
<td></td>
<td>Output HOMO, ..., HOMO-levels; LUMO, ..., LUMO+levels</td>
</tr>
<tr>
<td>trim</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dimerAB</td>
<td>AB.log</td>
<td></td>
<td>Log file of dimer AB</td>
</tr>
<tr>
<td>log</td>
<td>A.orb</td>
<td></td>
<td>Orbitals file</td>
</tr>
</tbody>
</table>

Return to the description of coupling.
4.2.2 log2mps
Generates an mps-file (with polar-site definitions) from a QM log-file

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>package</td>
<td></td>
<td></td>
<td>QM package</td>
</tr>
<tr>
<td>logfile</td>
<td></td>
<td></td>
<td>Log-file generated by QM package, with population/esp-fit data</td>
</tr>
</tbody>
</table>

Return to the description of log2mps.

4.2.3 molpol
Molecular polarizability calculator (and optimizer)

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mpsfiles</td>
<td></td>
<td></td>
<td>mps input file</td>
</tr>
<tr>
<td>input</td>
<td></td>
<td></td>
<td>mps output file</td>
</tr>
<tr>
<td>output</td>
<td></td>
<td></td>
<td>xml file with infos on polarizability tensor</td>
</tr>
<tr>
<td>polar</td>
<td></td>
<td></td>
<td>xml file with infos on polarizability tensor</td>
</tr>
<tr>
<td>induction</td>
<td></td>
<td></td>
<td>Thole sharpness parameter</td>
</tr>
<tr>
<td>expdamp</td>
<td></td>
<td></td>
<td>mixing factor for convergence</td>
</tr>
<tr>
<td>wSOR</td>
<td></td>
<td></td>
<td>maximum number of iterations</td>
</tr>
<tr>
<td>maxiter</td>
<td></td>
<td></td>
<td>rel. tolerance for induced moments</td>
</tr>
<tr>
<td>tolerance</td>
<td></td>
<td></td>
<td>if ‘true’, refine atomic polarizabilities to match molecular polarizable</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>if optimize=true the associated polarizable volume will be matched iteratively</td>
</tr>
<tr>
<td>target</td>
<td></td>
<td></td>
<td>relative tolerance when optimizing the polarizable volume</td>
</tr>
</tbody>
</table>

Return to the description of molpol.

4.2.4 pdb2map
Converts MD + QM files to VOTCA mapping. Combinations: pdb+xyz, gro+xyz, pdb

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pdb</td>
<td>conf.pdb</td>
<td></td>
<td>Input pdb file</td>
</tr>
<tr>
<td>gro</td>
<td>conf.gro</td>
<td></td>
<td>Input gro file</td>
</tr>
<tr>
<td>xyz</td>
<td>conf.xyz</td>
<td></td>
<td>Input xyz file</td>
</tr>
<tr>
<td>xml</td>
<td>conf.xml</td>
<td></td>
<td>Resulting xml file</td>
</tr>
</tbody>
</table>

Return to the description of pdb2map.

4.2.5 pdb2top
Generates fake Gromacs topology file .top

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>num</td>
<td>1</td>
<td></td>
<td>Num of mols in the box</td>
</tr>
<tr>
<td>pdb</td>
<td>conf.pdb</td>
<td></td>
<td>Input pdb file</td>
</tr>
</tbody>
</table>
4.2. CALCULATORS

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gro</td>
<td></td>
<td></td>
<td>Input gro file</td>
</tr>
<tr>
<td>conf.gro</td>
<td></td>
<td></td>
<td>Return to the description of pdb2top.</td>
</tr>
</tbody>
</table>

4.2.6 ptoreader

Reads binary .ptop-files (serialized from ewdgbpol) and processes them into something readable

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ptop_file</td>
<td></td>
<td></td>
<td>Binary archive .ptop-file</td>
</tr>
</tbody>
</table>

Return to the description of ptoreader.

4.2.7 eanalyze

Histogram and correlation function of site energies and pair energy differences

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolution_sites</td>
<td></td>
<td>eV</td>
<td>Bin size for site energy histogram</td>
</tr>
<tr>
<td>resolution_pairs</td>
<td></td>
<td>eV</td>
<td>Bin size for pair energy histogram</td>
</tr>
<tr>
<td>resolution_space</td>
<td></td>
<td>eV</td>
<td>Bin size for site energy correlation</td>
</tr>
<tr>
<td>states</td>
<td></td>
<td></td>
<td>?</td>
</tr>
</tbody>
</table>

Return to the description of eanalyze.

4.2.8 eimport

Imports site energies from the output file of emultipole and writes them to the state file

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
</table>

Return to the description of eimport.

4.2.9 einternal

Reads in site and reorganization energies and writes them to the state file

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>energiesXML</td>
<td></td>
<td></td>
<td>XML input file with vacuum site, reorganization (charging, discharging) energies</td>
</tr>
</tbody>
</table>

Return to the description of einternal.

4.2.10 emultipole

Evaluates polarization contribution based on the Thole model

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>multipoles</td>
<td></td>
<td></td>
<td>Polar Site Definitions in GDMA punch-file format</td>
</tr>
<tr>
<td>control</td>
<td></td>
<td></td>
<td>Control options for induction computation</td>
</tr>
<tr>
<td>induce</td>
<td>1</td>
<td></td>
<td>Enter '1' / '0' to toggle induction on / off</td>
</tr>
<tr>
<td>first</td>
<td></td>
<td></td>
<td>First segment for which to compute site energies</td>
</tr>
<tr>
<td>last</td>
<td></td>
<td></td>
<td>Last segment for which to compute site energies</td>
</tr>
<tr>
<td>output</td>
<td></td>
<td></td>
<td>File to write site energies to. Site energies are also stored in the state file</td>
</tr>
<tr>
<td>check</td>
<td></td>
<td></td>
<td>Check mapping of polar sites to fragment</td>
</tr>
</tbody>
</table>
## Thole Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tholeparam</td>
<td>Thole parameters required for charge-smearing</td>
</tr>
<tr>
<td>cutoff</td>
<td>Cut-off beyond which all interactions are neglected</td>
</tr>
<tr>
<td>cutoff2</td>
<td>Cut-off beyond which polarization is neglected</td>
</tr>
<tr>
<td>expdamp</td>
<td>Damping exponent used in exponential damping function</td>
</tr>
<tr>
<td>scaling</td>
<td>1-n interaction scaling, currently not in use</td>
</tr>
<tr>
<td>esp</td>
<td>Control options for potential calculation</td>
</tr>
<tr>
<td>calcESP</td>
<td>Enter ‘1’ / ‘0’ to toggle on / off. If ‘1’, site energies will not be evaluated</td>
</tr>
</tbody>
</table>

### Cube Grid

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cube</td>
<td>XYZ file specifying grid points for potential evaluation</td>
</tr>
<tr>
<td>grid</td>
<td>File to write grid-point potential to</td>
</tr>
</tbody>
</table>

### ESP Calculation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>calcESP</td>
<td>Enter ‘1’ / ‘0’ to toggle on / off. If ‘1’, site energies will not be evaluated</td>
</tr>
</tbody>
</table>

### Alpha Molecule Calculation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>calcAlpha</td>
<td>Enter ‘1’ / ‘0’ to toggle on / off. If ‘1’, site energies will not be evaluated</td>
</tr>
</tbody>
</table>

### Convergence Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>convparam</td>
<td>Convergence parameters for self-consistent field calculation</td>
</tr>
<tr>
<td>wSOR_N</td>
<td>Mixing factor for successive overrelaxation of neutral system, usually between 0.3 and 0.5</td>
</tr>
<tr>
<td>wSOR_C</td>
<td>Mixing factor for successive overrelaxation of charged system, usually between 0.3 and 0.5</td>
</tr>
<tr>
<td>tolerance</td>
<td>Convergence criterion, fulfilled if relative change smaller than tolerance</td>
</tr>
<tr>
<td>maxiter</td>
<td>Maximum number of iterations in the convergence loop</td>
</tr>
</tbody>
</table>

---

### Multipole Description

**4.2.11 eoutersphere**

Evaluates outersphere reorganization energy

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>multipoles</td>
<td></td>
<td>XML allocation polar sites</td>
<td></td>
</tr>
<tr>
<td>method</td>
<td></td>
<td>Type of the method: <strong>constant</strong> - all pairs have value <strong>lambda</strong>. <strong>spheres</strong> - molecules are treated as spheres with radii <strong>radius</strong> and Pekar factor <strong>pekar</strong>. <strong>dielectric</strong> - with Pekar factor <strong>pekar</strong> and partial charges from resulting dielectric fields</td>
<td></td>
</tr>
<tr>
<td>lambdacon</td>
<td>eV</td>
<td>The value for all pairs in the <strong>constant</strong> method</td>
<td></td>
</tr>
<tr>
<td>pekar</td>
<td></td>
<td>Pekar factor used for methods <strong>spheres</strong> and <strong>dielectric</strong></td>
<td></td>
</tr>
<tr>
<td>segment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>radius</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cutoff</td>
<td>nm</td>
<td>Cutoff radius in between pair and the exterior molecule. Can be used in <strong>spheres</strong> and <strong>dielectric</strong></td>
<td></td>
</tr>
</tbody>
</table>

---

1517 Return to the description of emultipole.

1518 **4.2.11 eoutersphere**

1519 Evaluates outersphere reorganization energy

1520 Return to the description of eoutersphere.
4.2.12 ewdbgpol
Calculates background polarisation needed for ewald calc

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>multipoles</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>control</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mps_table</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pdb_check</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>coulombmethod</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>method</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cutoff</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>shape</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>polarmethod</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>method</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>induce</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cutoff</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>convergence</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>energy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kfactor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rfactor</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Return to the description of ewdbgpol.

4.2.13 ianalyze
Evaluates a histogram of a logarithm of squared couplings

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>resolution_logJ2</td>
<td></td>
<td></td>
<td>Bin size of histogram log(J2)</td>
</tr>
<tr>
<td>states</td>
<td></td>
<td></td>
<td>States for which to calculate the histogram. Example: 1 -1</td>
</tr>
</tbody>
</table>

Return to the description of ianalyze.

4.2.14 iimport
Imports electronic couplings from xml of ctp-dipro using folders of pairdump

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>idft_jobs_file</td>
<td></td>
<td></td>
<td>idft jobs file</td>
</tr>
</tbody>
</table>

Return to the description of iimport.

4.2.15 izindo
Semiempirical electronic coupling elements for all neighbor list pairs

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orbitalsXML</td>
<td></td>
<td></td>
<td>File with paths to .orb files</td>
</tr>
</tbody>
</table>

Return to the description of izindo.

4.2.16 jobwriter
Writes list of jobs for a parallel execution

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 4. REFERENCE

<table>
<thead>
<tr>
<th>keys</th>
<th>states</th>
<th>single_id</th>
<th>kmc_cutoff</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>n e h</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>job type</td>
<td>hole, electron, neutral: mps file is required</td>
<td>Segment ID as argument for mps.single</td>
<td>Pair-interaction cut-off as argument for mps.kmc</td>
</tr>
</tbody>
</table>

Return to the description of jobwriter.

### 4.2.17 neighborlist

Constructs a list of neighboring conjugated segments

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>constant</td>
<td>0.5</td>
<td>nm</td>
<td>If provided, this value is used for all segment types</td>
</tr>
<tr>
<td>segments</td>
<td></td>
<td></td>
<td>A pair of segment types</td>
</tr>
<tr>
<td>type</td>
<td></td>
<td></td>
<td>Types of two segments. For types A and B this can be A</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A, A B or B B</td>
</tr>
<tr>
<td>cutoff</td>
<td></td>
<td>nm</td>
<td>Cutoff radius for centers of mass of rigid fragments</td>
</tr>
</tbody>
</table>

Return to the description of neighborlist.

### 4.2.18 pairdump

Coordinates of molecules and pairs from the neighbor list

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>molecules</td>
<td></td>
<td></td>
<td>If <strong>true</strong> outputs single molecules, otherwise only pairs</td>
</tr>
</tbody>
</table>

Return to the description of pairdump.

### 4.2.19 profile

Density and site energy profiles

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>axis</td>
<td></td>
<td></td>
<td>Axis along which to calculate density and energy profiles</td>
</tr>
<tr>
<td>direction</td>
<td>0 0 1</td>
<td></td>
<td>Axis direction</td>
</tr>
<tr>
<td>min</td>
<td></td>
<td>nm</td>
<td>Minimal projected position for manual binning</td>
</tr>
<tr>
<td>max</td>
<td></td>
<td>nm</td>
<td>Maximal projected position for manual binning</td>
</tr>
<tr>
<td>bin</td>
<td>0.1</td>
<td>nm</td>
<td>Spatial resolution of the profile</td>
</tr>
<tr>
<td>auto</td>
<td>1</td>
<td></td>
<td>'0' for manual binning using min and max, '1' for automated</td>
</tr>
<tr>
<td>particles</td>
<td></td>
<td></td>
<td>What centers of mass to use: 'segments' or 'atoms'</td>
</tr>
<tr>
<td>type</td>
<td>segments</td>
<td></td>
<td>ID of the first segment</td>
</tr>
<tr>
<td>first</td>
<td>1</td>
<td></td>
<td>ID of the first segment</td>
</tr>
<tr>
<td>last</td>
<td>-1</td>
<td></td>
<td>ID of the last segment, -1 is the list end</td>
</tr>
<tr>
<td>output</td>
<td>density.dat</td>
<td>density.dat</td>
<td>Density profile file</td>
</tr>
<tr>
<td></td>
<td>energy.dat</td>
<td>energy.dat</td>
<td>Energy profile file</td>
</tr>
</tbody>
</table>

Return to the description of profile.

### 4.2.20 rates

Hopping rates using classical or semi-classical expression

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
</table>
### 4.2. CALCULATORS

#### Field
- **field**
  - **temperature**: Field in x y z direction
  - **method**: Temperature for rates

#### Method
- **method**: Method chosen to compute rates. Can either be **marcus** or **jortner**. The first is the high temperature limit of Marcus theory, the second is the rate proposed by Jortner and Bixon

#### nmaxvib
- **nmaxvib**: If the method of choice is **jortner**, the maximal number of excited vibrations on the molecules has to be specified as an integer for the summation

#### Omega_vib
- **omegavib**: If the method of choice is **jortner**, the vibration frequency of the quantum mode has to be given in units of eV. The default value is close to the CC bond-stretch at 0.2eV

Return to the description of rates.

#### 4.2.21 sandbox

**sandbox**
- Sandbox to test ctp classes

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td></td>
<td></td>
<td>Not in use</td>
</tr>
</tbody>
</table>

Return to the description of sandbox.

#### 4.2.22 stateserver

**stateserver**
- Export SQLite file to human readable format

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>out</td>
<td></td>
<td></td>
<td>Output file name</td>
</tr>
<tr>
<td>pdb</td>
<td></td>
<td></td>
<td>PDB coordinate file name</td>
</tr>
<tr>
<td>keys</td>
<td></td>
<td></td>
<td>Sections to write to readable format (topology, segments, pairs, coordinates)</td>
</tr>
</tbody>
</table>

Return to the description of stateserver.

#### 4.2.23 tdump

**tdump**
- Coarse-grained and back-mapped (using rigid fragments) trajectories

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>md</td>
<td>MD.pdb</td>
<td></td>
<td>Name of the coarse-grained trajectory</td>
</tr>
<tr>
<td>qm</td>
<td>QM.pdb</td>
<td></td>
<td>Name of the trajectory with back-substituted rigid fragments</td>
</tr>
<tr>
<td>frames</td>
<td>1</td>
<td></td>
<td>Number of frames to output</td>
</tr>
</tbody>
</table>

Return to the description of tdump.

#### 4.2.24 vaverage

**vaverage**
- Computes site-centered velocity averages from site occupancies

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>carriers</td>
<td></td>
<td></td>
<td>Carrier types for which to compute velocity averages</td>
</tr>
<tr>
<td>tabulate</td>
<td></td>
<td></td>
<td>Tabulate ‘atoms’ or ‘segments’</td>
</tr>
</tbody>
</table>
4.2.25  zmultipole
Evaluates polarization contribution based on the Thole model

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>multipole</td>
<td></td>
<td></td>
<td>Polar Site Definitions in GDMA punch-file format</td>
</tr>
<tr>
<td>control</td>
<td></td>
<td></td>
<td>Control options for induction computation</td>
</tr>
<tr>
<td>induce</td>
<td>1</td>
<td></td>
<td>Enter ‘1’ / ‘0’ to toggle induction on / off</td>
</tr>
<tr>
<td>first</td>
<td></td>
<td></td>
<td>First segment for which to compute site energies</td>
</tr>
<tr>
<td>last</td>
<td></td>
<td></td>
<td>Last segment for which to compute site energies</td>
</tr>
<tr>
<td>output</td>
<td></td>
<td></td>
<td>File to write site energies to. Site energies are also stored</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>in the state file</td>
</tr>
<tr>
<td>check</td>
<td></td>
<td></td>
<td>Check mapping of polar sites to fragment</td>
</tr>
<tr>
<td>tholeparam</td>
<td></td>
<td></td>
<td>Thole parameters required for charge-smearing</td>
</tr>
<tr>
<td>cutoff</td>
<td>nm</td>
<td></td>
<td>Cut-off beyond which all interactions are neglected</td>
</tr>
<tr>
<td>cutoff2</td>
<td>nm</td>
<td></td>
<td>Cut-off beyond which polarization is neglected</td>
</tr>
<tr>
<td>expdamp</td>
<td></td>
<td></td>
<td>Damping exponent used in exponential damping function</td>
</tr>
<tr>
<td>scaling</td>
<td></td>
<td></td>
<td>1-n interaction scaling, currently not in use</td>
</tr>
<tr>
<td>esp</td>
<td></td>
<td></td>
<td>Control options for potential calculation</td>
</tr>
<tr>
<td>calcESP</td>
<td></td>
<td></td>
<td>Enter ‘1’ / ‘0’ to toggle on / off. If ‘1’, site energies will</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>not be evaluated</td>
</tr>
<tr>
<td>cube</td>
<td></td>
<td></td>
<td>XYZ file specifying grid points for potential evaluation</td>
</tr>
<tr>
<td>grid</td>
<td></td>
<td></td>
<td>File to write grid-point potential to</td>
</tr>
<tr>
<td>esf</td>
<td></td>
<td></td>
<td>Control options for field calculation</td>
</tr>
<tr>
<td>calcESF</td>
<td></td>
<td></td>
<td>Enter ‘1’ / ‘0’ to toggle on / off. If ‘1’, site energies will</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>not be evaluated</td>
</tr>
<tr>
<td>grid</td>
<td></td>
<td></td>
<td>XYZ file specifying grid points for field evaluation</td>
</tr>
<tr>
<td>output</td>
<td></td>
<td></td>
<td>File to write grid-point field to</td>
</tr>
<tr>
<td>alphamol</td>
<td></td>
<td></td>
<td>Control options for molecular-polarizability calculation</td>
</tr>
<tr>
<td>calcAlpha</td>
<td></td>
<td></td>
<td>Enter ‘1’ / ‘0’ to toggle on / off. If ‘1’, site energies will</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>not be evaluated</td>
</tr>
<tr>
<td>output</td>
<td></td>
<td></td>
<td>File to write polarizability tensor in global frame and in</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>diagonal form to</td>
</tr>
<tr>
<td>convparam</td>
<td></td>
<td></td>
<td>Convergence parameters for self-consistent field calculation</td>
</tr>
<tr>
<td>wSOR_N</td>
<td></td>
<td></td>
<td>Mixing factor for successive overrelaxation of neutral system,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>usually between 0.3 and 0.5</td>
</tr>
<tr>
<td>wSOR_C</td>
<td></td>
<td></td>
<td>Mixing factor for successive overrelaxation of charged system,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>usually between 0.3 and 0.5</td>
</tr>
<tr>
<td>tolerance</td>
<td></td>
<td></td>
<td>Convergence criterion, fulfilled if relative change smaller</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>than tolerance</td>
</tr>
<tr>
<td>maxiter</td>
<td></td>
<td></td>
<td>Maximum number of iterations in the convergence loop</td>
</tr>
</tbody>
</table>

4.2.26  edft
A wrapper for first principles based single site calculations

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>job</td>
<td></td>
<td></td>
<td>Job options</td>
</tr>
<tr>
<td>tasks</td>
<td>input,run,parse</td>
<td></td>
<td>What to run</td>
</tr>
<tr>
<td>store</td>
<td>orbitals</td>
<td></td>
<td>What to store</td>
</tr>
</tbody>
</table>
4.2. CALCULATORS

4.2.27 idft
Projection method for electronic couplings. Requires edft output

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tasks</td>
<td>input,run,parse</td>
<td></td>
<td>What to do</td>
</tr>
<tr>
<td>store</td>
<td>orbitals,overlap</td>
<td></td>
<td>What to store</td>
</tr>
<tr>
<td>degeneracy</td>
<td>0</td>
<td>eV</td>
<td>Criterium for the degeneracy of two levels</td>
</tr>
<tr>
<td>levels</td>
<td>3</td>
<td></td>
<td>Output between HOMO, ..., HOMO-levels; LUMO, ...</td>
</tr>
<tr>
<td>trim</td>
<td>2</td>
<td></td>
<td>Use trim*occupied of virtual orbitals</td>
</tr>
</tbody>
</table>

4.2.28 pewald3d
Evaluates site energies in a periodic setting

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>jobcontrol</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>job_file</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>multipoles</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mapping</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mps_table</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pdb_check</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>coulombmethod</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>method</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cutoff</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>shape</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>polarmethod</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>method</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>induce</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cutoff</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tasks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>calculate_fields</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>polarize_fg</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>evaluate_energy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>coarsegrain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cg_background</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cg_foreground</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cg_radius</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cg_anisotropic</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>convergence</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>energy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kfactor</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rfactor</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2.29 qmmm
QM/MM with the Thole MM model

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
</table>

Return to the description of edft.

Return to the description of idft.

Return to the description of pewald3d.
### CHAPTER 4. REFERENCE

<table>
<thead>
<tr>
<th>control</th>
<th>pdb_check</th>
<th>PDB file of polar sites</th>
</tr>
</thead>
<tbody>
<tr>
<td>write_chk</td>
<td>dipoles.xyz</td>
<td>XYZ file with dipoles split onto point charges</td>
</tr>
<tr>
<td>format_chk</td>
<td>xyz</td>
<td>format, gaussian or xyz</td>
</tr>
<tr>
<td>split_dpl</td>
<td>1</td>
<td>‘0’ do not split dipoles onto point charges, ‘1’ do split</td>
</tr>
<tr>
<td>dpl_spacing</td>
<td>1e-3</td>
<td>nm</td>
</tr>
<tr>
<td>qmpackage</td>
<td>package</td>
<td>QM package to use for the QM region</td>
</tr>
<tr>
<td>gwbse</td>
<td>gwbse_options</td>
<td>Specify if GW/BSE excited state calculation ist needed</td>
</tr>
<tr>
<td>state</td>
<td></td>
<td>GW/BSE options file</td>
</tr>
<tr>
<td>type</td>
<td></td>
<td>Number of excited state, which is to be calculated</td>
</tr>
<tr>
<td>filter</td>
<td></td>
<td>Character of the excited state to be calculated</td>
</tr>
<tr>
<td>oscillator_strength</td>
<td></td>
<td>Filter with which to find the excited state after each calculation</td>
</tr>
<tr>
<td>charge_transfer</td>
<td></td>
<td>Oscillator strength filter, only states with higher oscillator strength are considered</td>
</tr>
<tr>
<td>qmmmconvg</td>
<td>dR</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>dQ</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>dE_QM</td>
<td>0.0001</td>
</tr>
<tr>
<td></td>
<td>dE_MM</td>
<td>0.0001</td>
</tr>
<tr>
<td></td>
<td>max_iter</td>
<td>10</td>
</tr>
<tr>
<td>coulombmethod</td>
<td>method</td>
<td></td>
</tr>
<tr>
<td>cutoff1</td>
<td></td>
<td>cut-off</td>
</tr>
<tr>
<td>cutoff2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tholemodel</td>
<td>induce</td>
<td></td>
</tr>
<tr>
<td></td>
<td>induce_intra_pair</td>
<td></td>
</tr>
<tr>
<td>exp_damp</td>
<td>scaling</td>
<td>0.39</td>
</tr>
<tr>
<td>convergence</td>
<td></td>
<td></td>
</tr>
<tr>
<td>wSOR_N</td>
<td></td>
<td></td>
</tr>
<tr>
<td>wSOR_C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>max_iter</td>
<td>512</td>
<td>Maximal number of iterations to converge induced dipoles</td>
</tr>
<tr>
<td>tolerance</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Return to the description of qmmm.

### 4.2.30 xqmultipole

Electrostatic interaction and induction energy of charged molecular clusters

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>multipoles</td>
<td></td>
<td></td>
<td>Polar-site mapping definition</td>
</tr>
<tr>
<td>control</td>
<td>job_file</td>
<td></td>
<td>Job file</td>
</tr>
<tr>
<td></td>
<td>emp_file</td>
<td></td>
<td>Polar-background definition, allocation of mps-files to segments</td>
</tr>
</tbody>
</table>
### 4.2. Calculators

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pdb_check</td>
<td>Whether or not to output a pdb-file of the mapped polar sites</td>
</tr>
<tr>
<td>format_chk</td>
<td>Format for check-file: 'xyz' or 'gaussian'</td>
</tr>
<tr>
<td>split_dpl</td>
<td>Split dipoles onto point charges in check-file</td>
</tr>
<tr>
<td>dpl_spacing</td>
<td>nm Spacing between point charges for check-file output</td>
</tr>
<tr>
<td>coulombmethod</td>
<td>method</td>
</tr>
<tr>
<td></td>
<td>Currently only cut-off supported</td>
</tr>
<tr>
<td>cutoff1</td>
<td>nm Full-interaction radius cut-off</td>
</tr>
<tr>
<td>cutoff2</td>
<td>nm Radius of electrostatic buffer</td>
</tr>
<tr>
<td>tholemodel</td>
<td>induce</td>
</tr>
<tr>
<td></td>
<td>Induce - or not</td>
</tr>
<tr>
<td>induce_intra_pair</td>
<td>Induce mutually within the charged cluster</td>
</tr>
<tr>
<td>exp_damp</td>
<td>Thole sharpness parameter</td>
</tr>
<tr>
<td>scaling</td>
<td>Bond scaling parameters, currently not used</td>
</tr>
<tr>
<td>convergence</td>
<td>wSOR_N</td>
</tr>
<tr>
<td></td>
<td>SOR mixing factor for overall neutral clusters</td>
</tr>
<tr>
<td>wSOR_C</td>
<td>SOR mixing factor for overall charged clusters</td>
</tr>
<tr>
<td>max_iter</td>
<td>Maximum number of iterations</td>
</tr>
<tr>
<td>tolerance</td>
<td>Relative tolerance as convergence criterion</td>
</tr>
</tbody>
</table>

Return to the description of xqmultipole.

### 4.2.31 energy2xml

Write out energies from SQL file

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
</table>

Return to the description of energy2xml.

### 4.2.32 integrals2xml

Write out transfer integrals from SQL file

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
</table>

Return to the description of integrals2xml.

### 4.2.33 occupations2xml

Write out site occupation probabilities from SQL file

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
</table>

Return to the description of occupations2xml.

### 4.2.34 pairs2xml

Write out neighbourlist from SQL file

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
</table>

Return to the description of pairs2xml.
4.2.35  **rates2xml**
Write out charge transfer rates from SQL file

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
</table>

Return to the description of **rates2xml**.

4.2.36  **segments2xml**
Write out segment data from SQL file

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
</table>

Return to the description of **segments2xml**.

4.2.37  **trajectory2pdb**
Generate PDB files for the mapped MD/QM topology

<table>
<thead>
<tr>
<th>option</th>
<th>default</th>
<th>unit</th>
<th>description</th>
</tr>
</thead>
</table>

Return to the description of **trajectory2pdb**.
Bibliography

[34] V. Coropceanu et al., Chemical Reviews 107, 926 (2007).