potter

1.0

Generated by Doxygen 1.8.17

Chapter 1

README

Eigen is a C++ template library for linear algebra: matrices, vectors, numerical solvers, and related algorithms.

For more information go to http://eigen.tuxfamily.org/.

For ***pull request*** please only use the official repository at https://bitbucket.org/eigen/eigen.

For ***bug reports*** and ***feature requests*** go to http://eigen.tuxfamily.org/bz.

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Class Documentation

3.1 Integrator< TYPE > Class Template Reference

Public Types

- using **EColArray** = Eigen::Array< TYPE, Eigen::Dynamic, 1 >

Public Member Functions

- **Integrator** (const Molecule< TYPE > &mol1, const Molecule< TYPE > &mol2)
- template<typename TEMPTYPE >
  TEMPTYPE **radial_integrate_B2** (TEMPTYPE Tstar, TYPE rstart, TYPE rend, int N)
- auto & **get_evaluator** ()
- void **init_thread_pool** (short Nthreads)
- double **potential** (double r, double theta1, double theta2, double phi)
- TYPE **orient_averaged_potential** (TYPE rstar, Molecule< TYPE > mol1, Molecule< TYPE > mol2) const
- template<typename TEMPTYPE >
  std::tuple< TEMPTYPE, TEMPTYPE > one_temperature (int order, TEMPTYPE Tstar, TYPE rstart, TYPE rend, Molecule< TYPE > mol1, Molecule< TYPE > mol2) const
- std::map< std::string, double > **B_and_derivs** (int order, int Nderivs, double T, double rstart, double rend, Molecule< TYPE > mol1, Molecule< TYPE > mol2)
- auto **parallel_B_and_derivs** (int order, int Nthreads, int Nderivs, std::vector< double > Tvec, double rstart, double rend, Molecule< TYPE > mol1, Molecule< TYPE > mol2)
- TYPE **orientation_averaged_integrate** (const TYPE Tstar, const TYPE rstart, const TYPE rend)

Public Attributes

- const Molecule< TYPE > **mol1**
- const Molecule< TYPE > **mol2**
- PotentialEvaluator< TYPE > **potcls**

3.1.1 Detailed Description

template<typename TYPE>
class Integrator< TYPE >

Definition at line 410 of file potter.hpp.

3.1.2 Member Function Documentation

3.1.2.1 one_temperature()

```
template<typename TYPE >
template<typename TEMPTYPE >
std::tuple<TEMPTYPE,TEMPTYPE> Integrator< TYPE >::one_temperature (
            int order,
            TEMPTYPE Tstar,
            TYPE rstart,
            TYPE rend,
            Molecule< TYPE > mol1,
            Molecule< TYPE > mol2 ) const  [inline]
```

Do the calculations for one temperature

Parameters

order	The order of the virial coefficient (2=B_2, 3=B_3, etc.)
Tstar	The temperature
rstart	The initial value of r to be considered in integration
rend	The final value of r to be considered in integration
mol1	The first molecule
mol2	The second molecule

Returns

Tuple of (value, estimated error in value)

Note

The return numerical type maybe be one of double, std::complex<double> or MultiComplex<double>

Definition at line 504 of file potter.hpp.

The documentation for this class was generated from the following file:

- include/potter/potter.hpp

3.2 Molecule< TYPE > Class Template Reference

Public Types

- using **EColArray** = Eigen::Array< TYPE, Eigen::Dynamic, 1 >
- using **CoordMatrix** = Eigen::Array< TYPE, 3, Eigen::Dynamic >

Public Member Functions

- **Molecule** (const std::vector< std::vector< TYPE >> &pts)
- void **reset** ()
- CoordMatrix **rotZ3** (TYPE theta) const
- CoordMatrix **rotY3** (const TYPE theta) const
- CoordMatrix **rotX3** (TYPE theta) const
- void **rotate_plusx** (TYPE angle)
- void **rotate_negativex** (TYPE angle)
- void **rotate_plusy** (TYPE angle)
- void **rotate_negativey** (TYPE angle)
- void **rotate_plusz** (TYPE angle)
- void **rotate_negativez** (TYPE angle)
- void **translatex** (TYPE dx)
- void **translatey** (TYPE dy)
- template<typename ArrayType >
  TYPE **get_dist** (const ArrayType &rA, const ArrayType &rB) const
- Eigen::Index **get_Natoms** () const
- auto **get_xyz_atom** (const Eigen::Index i) const

Public Attributes

- CoordMatrix **coords**
- CoordMatrix **coords_initial**

3.2.1 Detailed Description

template<typename TYPE>
class Molecule< TYPE >

Definition at line 71 of file potter.hpp.

The documentation for this class was generated from the following file:

- include/potter/potter.hpp

3.3 PotentialEvaluator< TYPE > Class Template Reference

Public Member Functions

- TYPE **eval_pot** (const Molecule< TYPE > &molA, const Molecule< TYPE > &molB) const
- void **connect_potentials** (std::function< double(double)> &f, std::size_t Natoms)
- void **add_potential** (std::size_t iatom, std::size_t jatom, std::function< double(double)> &f)
- auto & **get_potential** (std::size_t i, std::size_t j) const

Public Attributes

- std::map< std::tuple< std::size_t, std::size_t >, std::function< double(double)> > **potential_map**

3.3.1 Detailed Description

template<typename TYPE>
class PotentialEvaluator< TYPE >

Definition at line 157 of file potter.hpp.

The documentation for this class was generated from the following file:

- include/potter/potter.hpp

3.4 SharedDataBase< TYPE, TEMPTYPE > Class Template Reference

A helper class.

`#include <potter.hpp>`

Public Member Functions

- **SharedDataBase** (TEMPTYPE Tstar, TYPE rstar, Molecule< TYPE > molA, Molecule< TYPE > molB, const PotentialEvaluator< TYPE > &evaltr, const std::valarray< TYPE > &xmin, const std::valarray< TYPE > &xmax)
- TYPE **eval_pot** (const Molecule< TYPE > &molA, const Molecule< TYPE > &molB)
- void **orient_integrand** (double theta1, double theta2, double phi, double *fval)
- void **atomic_B3_integrand** (const double r12, const double r13, const double eta_angle, double *fval)
- void **atomic_B4_1_integrand** (const double r14, const double r13, const double gamma_angle, const double r12, const double eta_angle, double *fval)
- void **atomic_B4_2_integrand** (const double eta_angle, const double r12, const double r13, const double gamma_angle, const double r14, double *fval)
- void **atomic_B4_3_integrand** (const double eta_angle, const double zeta_angle, const double gamma_↩ angle, const double r12, const double r13, const double r14, double *fval)

Public Attributes

- TEMPTYPE **Tstar**
- TYPE **rstar**
- Molecule< TYPE > **molA**
- Molecule< TYPE > **molB**
- std::valarray< TYPE > **xmin**
- std::valarray< TYPE > **xmax**
- const PotentialEvaluator< TYPE > & **evaltr**

3.4.1 Detailed Description

template<typename TYPE, typename TEMPTYPE>
class SharedDataBase< TYPE, TEMPTYPE >

A helper class.

Definition at line 212 of file potter.hpp.

The documentation for this class was generated from the following file:

- include/potter/potter.hpp

Index