
OpenRSP Documentation

Release 1.0.0-alpha

OpenRSP authors

Nov 20, 2018

| | | |
|-----------|---|-----------|
| 1 | What is OpenRSP? | 3 |
| 2 | Authors | 5 |
| 3 | Citation guide | 7 |
| 4 | A brief history | 9 |
| 5 | Programs where OpenRSP is used | 11 |
| 6 | Papers involving OpenRSP | 13 |
| 7 | Theoretical background | 15 |
| 8 | Get and run OpenRSP | 17 |
| 9 | Add OpenRSP to a quantum chemistry program | 19 |
| 10 | Get involved with development | 21 |
| 11 | How Sphinx works | 23 |
| 12 | Tentative Rules for Developers | 25 |
| | Bibliography | 27 |

Welcome to the website of OpenRSP - a program library for the open-ended, analytic calculation of molecular properties! Please choose a topic to learn more about what OpenRSP is, who is involved in it, how you can use it or how you can get involved.

What is OpenRSP?

OpenRSP is a program library that uses recursive routines to identify and assemble contributions to response properties - that is, molecular properties as they are expressed in the theory called “response theory” from theoretical chemistry.

The name of OpenRSP reflects the following features:

- It is a library for the **Open**-ended calculation of **ReSP**onse properties: It can be used for the calculation of response properties to arbitrary order.
- It is **Open**-source and is publicly available under the LGPL v2.1 license.
- It has an application programming interface that **Opens** it to connection with other programs that wish to make use of its functionality.

1.1 What are response properties?

Response properties describe how fundamental properties of a molecular system *respond* to external influences like subjection to an electromagnetic field or displacement of the atomic nuclei. They and related properties are essential for the description of spectroscopic processes and molecular characteristics like infrared spectroscopy, Raman scattering, multiphoton absorption and vibrational energy levels. If you have ever done computational work on the molecular level for phenomena in this category, chances are that response properties were involved at some stage of the calculation.

Response properties can be categorized by their *order*, that is, the “number of influences” that were taken into consideration for a given property. The first such order is called *linear response* and contains much-used properties like the electric dipole polarizability - i.e. the first-order change to the molecular dipole moment in the presence of an electric field - or the Hessian matrix of nuclear geometric displacements - i.e. the change in the molecular gradient that would result from displacing each coordinate of the molecular geometry.

Higher orders of response properties describe the changes that the fundamental molecular property would undergo upon subjection to more than one external influence, or upon higher-order interactions with the same influence. Examples of such properties are the geometric gradient of the electric dipole polarizability - essential for the description of vibrational Raman spectra - or the cubic and quartic force constants, i.e. the third- and fourth-order derivatives of the molecular energy with respect to geometrical displacements - which may be used to calculate corrections to a description of the vibrational energy levels stemming from the geometric Hessian.

1.2 Why use OpenRSP?

By its recursive structure, OpenRSP makes it possible to calculate response properties of arbitrary complexity in an analytical manner, not resorting to numerical schemes like finite difference methods in the calculation. Compared to analytical methods, numerical approaches may be associated with a greater degree of uncertainty related to accuracy and practical feasibility of the calculation, and we therefore think that analytical calculation should be used whenever it is practical.

Today's programs written for the calculation of response properties may either not have a recursive structure, or may use numerical methods to different extents, or both. In the cases where existing programs use an analytical approach, they may either be not recursive (which typically means that a new program routine must be written for each new property for which calculation is desired), or they may only be usable for a limited category of properties. As the complexity of the expressions that must be evaluated in an analytical approach to yield the desired response property increases rapidly with the order of response, such analytic calculation of high-order response properties can quickly become a very complicated task and the implementation of *ad hoc* program routines for their calculation may be intractable at higher orders.

The structure of OpenRSP, using recursion as a core tool, solves the task of identifying and assembling contributions to response properties "once and for all". When combined with program libraries that can provide the contributions that OpenRSP identifies, any response property can be calculated fully analytically as long as those libraries can provide the necessary contributions. We note, however, that the present version of the code is still awaiting the completion of functionality to handle perturbations that both change the basis set and have a nonzero frequency associated with them, but that such extension is within the scope of the present underlying theory.

This table lists the main developers of OpenRSP and their current affiliation:

Table 1: OpenRSP authors (alphabetical order of surname)

| Name | Affiliation |
|------------------------|---------------------------------------|
| Radovan Bast | UiT The Arctic University of Norway |
| Daniel H. Friese | Heinrich-Heine-Universität Düsseldorf |
| Bin Gao | UiT The Arctic University of Norway |
| Dan J. Jonsson | UiT The Arctic University of Norway |
| Magnus Ringholm | UiT The Arctic University of Norway |
| Simen S. Reine | University of Oslo |
| Kenneth Ruud | UiT The Arctic University of Norway |
| Andreas J. Thorvaldsen | Science [&] Technology AS |

Requests or comments should primarily be directed to authors listed in **boldface** whose e-mail addresses are all in the format *firstname.lastname@uit.no*.

All published results obtained with OpenRSP are expected to cite the following references:

- Andreas J. Thorvaldsen, Kenneth Ruud, Kasper Kristensen, Poul Jørgensen and Sonia Coriani, *J. Chem. Phys.* **129**, 214108 (2008).
- Magnus Ringholm, Dan Jonsson and Kenneth Ruud, *J. Comput. Chem.* **35**, 622-633 (2014).
- [OpenRSP] OpenRSP, an open-ended response property library, www.openrsp.org

If the use of OpenRSP involved the calculation of single residues of response properties, citation of the following reference is expected in addition to the references listed above:

- Daniel H. Friese, Maarten T. P. Beerepoot, Magnus Ringholm and Kenneth Ruud, *J. Chem. Theory Comput.* **11**, 1129-1144 (2015).

Please also note that host programs into which OpenRSP is incorporated may have their own citation guidelines or requirements to be observed if such programs are used.

4.1 The OpenRSP core functionality

Work on the OpenRSP project began in the mid-2000's when the first work started on what has become the present version of the program. During this time, the theoretical foundation of response theory on which OpenRSP was based was developed and used to create program routines that were connected to the Dalton quantum chemistry program. This version of the code was used to compute several response properties for which analytic calculation had not been carried out before.

In 2008, we have generalized OpenRSP for the DIRAC program package which enabled us to access a wealth of response properties at the 4-component relativistic level.

In 2011, work was started on a version - then still a part of Dalton - where recursion was used to achieve an open-ended implementation of the theory, so that one set of routines could be used to manage the calculation of any response property. This version forms the basis of the present-day core functionality of the OpenRSP, but was since developed further to include features such as calculation of single residues of response properties (of use in the calculation of multiphoton strengths), calculation of multiple properties in one invocation with reuse of common intermediate results, and restructuring of calls to external routines to reduce recalculation of various contributions such as perturbed one- and two-electron integrals.

4.2 OpenRSP as a modular library with an API

In order to make OpenRSP into a modular library that was not tied to any one particular quantum chemistry program - or *host program* - work began in 2013 on developing an application programming interface (API) for OpenRSP, involving the creation of clearly defined interfaces between OpenRSP and other codes, the use of callback routines in order to abstract the way in which the OpenRSP core asks for contributions from external libraries, and the development of the QcMatrix library to abstract and mediate matrix operations so that OpenRSP is agnostic to the underlying implementation of such operations. The first host program to make use of this modular functionality is the [LSDalton](#) quantum chemistry program.

4.3 Libraries for external contributions

During the course of its execution, OpenRSP identifies various contributions that it must get from libraries external to it in order to be able to assemble the response property or properties to be calculated, such as perturbed one- and two-electron integral contributions, exchange-correlation contributions if a density-functional theory calculation is requested, or solution of so-called response equations. Therefore, the development of libraries that can provide such functionality at a sufficient level of generality - although not necessarily driven by the demands of OpenRSP - has nevertheless been an important concurrent task, and has resulted in the creation of sophisticated software without which OpenRSP would not be able to do what it does best. Some of the libraries that are presently used or have been used by OpenRSP are listed below:

- Gen1Int for the calculation of perturbed one-electron integrals
- cgto-diff-eri for the calculation of perturbed two-electron integrals
- HODI for the calculation of perturbed integrals
- XCint and XCfun for the calculation of exchange-correlation contributions
- A linear response equation solver by Sonia Coriani et al.
- FraME for a polarizable embedding description of molecular surroundings
- PCMSolver for a polarizable continuum description of molecular surroundings

Programs where OpenRSP is used

The following programs feature or have featured OpenRSP in some version:

[Dalton](#) has featured OpenRSP in a private version at an earlier stage of development, but that version of OpenRSP is now outdated.

[LSDalton](#) will soon feature a new and public version of OpenRSP that is in active development.

Papers involving OpenRSP

This is a list of scientific articles where OpenRSP is involved in some capacity, either pertaining to theoretical development related to the core functionality or related functionality, or as having been applied to produce computational results.

6.1 2018

6.2 2017

6.3 2016

6.4 2015

6.5 2014

6.6 2008

Theoretical background

We are working on a documentation of the OpenRSP core routines and its application programming interface (API), and this, together with a introduction of the underlying theory of OpenRSP intended to be accessible, will be made available on this website once ready.

In the meantime, for a technical explanation of the theoretical background of OpenRSP, the following references may prove informative:

The paper describing the version of response theory upon which OpenRSP is based.

- Andreas J. Thorvaldsen, Kenneth Ruud, Kasper Kristensen, Poul Jørgensen and Sonia Coriani, *J. Chem. Phys.* **129**, 214108 (2008)

Describes a recursive algorithmic approach for the calculation of response properties:

- Magnus Ringholm, Dan Jonsson and Kenneth Ruud, *J. Comput. Chem.* **35**, 622-633 (2014)

Describes a recursive algorithmic approach for the calculation of single residues of response functions that can be used to obtain multiphoton absorption matrix elements:

- Daniel H. Friese, Maarten T. P. Beerepoot, Magnus Ringholm and Kenneth Ruud, *J. Chem. Theory Comput.* **11**, 1129-1144 (2015)

Describes a recursive algorithmic approach for the calculation of single residues of response functions that contain perturbations which affect the basis set (please note that this functionality is not yet implemented in the latest version of OpenRSP):

- Daniel H. Friese, Magnus Ringholm, Bin Gao and Kenneth Ruud, *J. Chem. Theory Comput.* **11** (10), 4814 (2015)

Get and run OpenRSP

If you want to get OpenRSP and use it for calculations, then please make note of the following: OpenRSP is a program library that manages the calculation of response properties, and it **cannot** calculate these properties without getting contributions like perturbed one- and two-electron integrals or solutions of response equations from other codes to which it connects through the application programming interface (API). This means that if you download and build OpenRSP from its [GitHub repository](#), the compiled product will not on its own be able to calculate response properties. A set of API connections to enable OpenRSP to manage response property calculations can for example be made in quantum chemistry programs where necessary routines for these contributions are implemented.

Consequently, in order to use OpenRSP for calculations, it is necessary to use it in a *host program* into which OpenRSP has been incorporated in this way, and a list of such programs is kept at the [Programs where OpenRSP is used](#) page. The specific way in which OpenRSP is invoked in a host program - i.e. the way that you can make OpenRSP calculate something in that program - is a feature of each such program, and you must therefore follow the relevant instructions to achieve this, as may for example be shown in the user manual for the host program that you want to use.

Add OpenRSP to a quantum chemistry program

If you want to add OpenRSP to a quantum chemistry program, then you are free to do so provided that you do not violate OpenRSP's LGPL v2.1 software license as described on OpenRSP's [GitHub repository](#).

In order to enable OpenRSP to work as intended, you must provide routines that connect to the OpenRSP application programming interface (API) to give OpenRSP access to contributions such as perturbed one- and two electron integrals, exchange-correlation contributions if calculations at the density-functional theory (DFT) level is desired, or solution routines for response equations. Please note that OpenRSP is a program library that manages the calculation of response properties, and it **cannot** carry out actual such calculations without getting contributions like the ones mentioned here from program routines that are external to OpenRSP.

A description of the API is under development and will be made available on this website. In the meantime, questions may be directed to the *Authors*.

CHAPTER 10

Get involved with development

We welcome your participation if you want to become involved with the development of OpenRSP! Our code is [hosted on GitHub](#) and is publicly available under the LGPL v2.1 software license. You may freely obtain and use this code provided that you do not violate this software license, but us present *Authors* would of course would of course also like to get in touch with you.

We are still working on a documentation of the OpenRSP code and API, and this will be made available on this website when ready. A style guide and contribution guidelines are also under development.

These pages are generated using Sphinx. If you want to find out more about RST/Sphinx, please read <http://sphinx-doc.org/rest.html>. RST is a subset of Sphinx. Sphinx is RST with some extensions.

11.1 How to modify the website

The website is generated from RST sources under `doc/`. Once a pull request is merged, a post-receive hook updates the documentation on <https://openrsp.readthedocs.io>. This typically takes less than a minute. Our main page <http://openrsp.org> redirects to <https://openrsp.readthedocs.io>.

11.2 How to locally test changes

You don't have to push to see and test your changes. You can test them locally. For this install the Python packages `sphinx` and `sphinx_rtd_theme`. Then build the pages with:

```
$ sphinx-build doc/ _build
```

Then point your browser to `_build/html/index.html`. The style is not the same but the content is what you would see after a successful pull request merge.

Tentative Rules for Developers

12.1 Short-version

1. First analyze the problem, then design the code (data and algorithm structures), prepare test suite. Last, write the code.
2. Make everything as simple as possible.
3. Do your best to prepare a readable document.

12.2 Long-version

1. First of all, please write explicitly what you would like to implement in `doc!` Describe your idea using formulas and/or words. Then translate them into algorithms and data structure. Please do write what objects/types/variables you will define and their corresponding public and private functions (including detailed descriptions of the input and output arguments). It would be better if you could write down the framework of your implementation using figures. Please also write down the limitations or risks of your code, for instance, does it stable or have some numerical error? If yes, how to prevent or how to know if the results are reasonable?

In this stage, you may refer to some rules in object-oriented programming (OOP). For instance, when you define a module/class etc.:

1. it should be open for extension but closed for modification (Open Closed Principle, OCP),
 2. subclasses should be substitutable for their base classes (Liskov Substitution Principle, LSP),
 3. depend upon abstractions, do not depend upon concretions (Dependency Inversion Principle, DIP),
 4. many client specific interfaces are better than one general purpose interface (Interface Segregation Principle, ISP),
 5. In other words: low coupling, high cohesion, open for extension, and closed for changes (from “Developing Chemical Information Systems: An Object-Oriented Approach Using Enterprise Java”, Fan Li).
2. Write the codes. During this stage, we would be happy if you could:

1. write comments (in english, one line for each 10-20 line of codes at least),
 2. try to use descriptive names for your classes and methods,
 3. do your best to avoid global variables,
 4. try to re-use code and try to use libraries,
3. This is very important, and should be considered and implemented during the aforementioned two steps:
- Always provide a test suite for each function/subroutine/module etc., unless you are 100% sure what you did is right. Integration testing will also be required in some cases.

Bibliography

- [Morgan2018] Geometric Energy Derivatives at the Complete Basis Set Limit: Application to the Equilibrium Structure and Molecular Force Field of Formaldehyde, Morgan, W. James; Matthews, Devin A.; Ringholm, Magnus; et al. *J. Chem. Theory Comput.* 14 (3), 1333 (2018)
- [DiRemigio2017] Open-ended formulation of self-consistent field response theory with the polarizable continuum model for solvation Di Remigio, Roberto; Beerepoot, Maarten T. P.; Cornaton, Yann; et al. *PCCP* 19 (1), 366 (2017)
- [Anelli2017] Gauge-origin independent calculations of electric-field-induced second-harmonic generation circular intensity difference using London atomic orbitals Anelli, Marco; Ringholm, Magnus; Ruud, Kenneth *Mol. Phys.* 115 (1-2), 241 (2017)
- [Steindal2016] Open-ended response theory with polarizable embedding: multiphoton absorption in biomolecular systems Steindal, Arnfinn Hykkerud; Beerepoot, Maarten T. P.; Ringholm, Magnus; et al. *PCCP* 18 (40), 28339 (2016)
- [Cornaton2016-2] Complete analytic anharmonic hyper-Raman scattering spectra Cornaton, Yann; Ringholm, Magnus; Ruud, Kenneth *PCCP* 18 (32), 22331 (2016)
- [Cornaton2016] Analytic calculations of anharmonic infrared and Raman vibrational spectra Cornaton, Yann; Ringholm, Magnus; Louant, Orian; et al. *PCCP* 18 (5) 4201 (2016)
- [Friese2015-2] Open-Ended Recursive Calculation of Single Residues of Response Functions for Perturbation-Dependent Basis Sets Friese, Daniel H.; Ringholm, Magnus; Gao, Bin; et al. *J. Chem. Theory Comput.* 11 (10), 4814 (2015)
- [Friese2015] Open-Ended Recursive Approach for the Calculation of Multiphoton Absorption Matrix Elements Friese, Daniel H.; Beerepoot, Maarten T. P.; Ringholm, Magnus; et al. *J. Chem. Theory Comput.* 11 (3), 1129 (2015)
- [Ringholm2014-3] Analytic calculations of hyper-Raman spectra from density functional theory hyperpolarizability gradients Ringholm, Magnus; Bast, Radovan; Oggioni, Luca; et al. *J. Chem. Phys.* 141 (13), 134107 (2014)
- [Ringholm2014-2] Analytic cubic and quartic force fields using density-functional theory Ringholm, Magnus; Jonsson, Dan; Bast, Radovan; et al. *J. Chem. Phys.* 140 (3), 034103 (2014)
- [Gao2014] Analytic Density Functional Theory Calculations of Pure Vibrational Hyperpolarizabilities: The First Dipole Hyperpolarizability of Retinal and Related Molecules Gao, Bin; Ringholm, Magnus; Bast, Radovan; et al. *J. Phys. Chem. A* 118 (4), 748 (2014)

[Ringholm2014] A General, Recursive, and Open-Ended Response Code Ringholm, Magnus; Jonsson, Dan; Ruud, Kenneth J. *Comput. Chem.* 35 (8), 622 (2014)

[Thorvaldsen2008] A density matrix-based quasienergy formulation of the Kohn–Sham density functional response theory using perturbation- and time-dependent basis sets Thorvaldsen, Andreas J.; Ruud, Kenneth; Kristensen, Kasper; et al. *J. Chem. Phys.* 129 (21), 214108 (2008)