OpenQP: A Quantum Chemical Platform Featuring MRSF-TDDFT with an Emphasis on Open-source Ecosystem

Vladimir Mironov,[†] Konstantin Komarov,[‡] Jingbai Li,[¶] Igor Gerasimov,[†] Hiroya Nakata,[§] Mohsen Mazaheri,[†] Kazuya Ishimura,[‖] Woojin Park,[†] Alireza Lashkaripour,[†] Minseok Oh,[⊥] Miquel Huix-Rotllant,[#] Seunghoon Lee,[⊥] and Cheol Ho Choi[*,†]

†*Department of Chemistry, Kyungpook National University, Daegu 41566, South Korea*

‡*Center for Quantum Dynamics, Pohang University of Science and Technology, Pohang 37673, South Korea*

¶*Hoffmann Institute of Advanced Materials, Shenzhen, China*

§*Fukui Institute for Fundamental Chemistry, Kyoto University, Kyoto 606-8103, Japan*

‖*X-Ability Co., Ltd., Ishiwata Building 3rd Floor, 4-1-5 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan*

⊥*Department of Chemistry, Seoul National University, Seoul 151-747, South Korea*

#*Aix-Marseille Univ, CNRS, ICR, Marseille 13397, France*

E-mail: cchoi@knu.ac.kr

Abstract

The **OpenQP** (Open Quantum Platform) is a new open-source quantum chemistry library developed to tackle sustainability and interoperability challenges in the field of computational chemistry. **OpenQP** provides various popular quantum chemical theories as autonomous **modules** such as energy and gradient calculations of HF, DFT, TDDFT, SF-TDDFT, and MRSF-TDDFT, thereby allowing easy interconnection with third-party software. A scientifically notable feature is the innovative Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory (MRSF-TDDFT) and its customized exchange-correlation functionals such as DTCAM series of VAEE, XI, XIV, AEE, and VEE, which significantly expands the applicability scope of DFT and TDDFT. **OpenQP** also supports parallel execution and is optimized with BLAS and LAPACK for high performance. Future enhancements such as EKT-MRSF-TDDFT and SOC-MRSF-TDDFT will further expand **OpenQP**'s capabilities. Additionally, a Python wrapper **PyOQP** is provided that performs tasks such as geometry optimization, conical intersection searches, and nonadiabatic coupling calculations, among others, by prototyping the **modules** of **OpenQP** library in combination with third-party libraries.

Overall, **OpenQP** aligns with modern trends in high-performance scientific software development by offering flexible prototyping and operation while retaining the performance benefits of compiled languages like Fortran and C. They enhance the sustainability and interoperability of quantum chemical software, making **OpenQP** a crucial platform for accelerating the development of advanced quantum theories like MRSF-TDDFT.

1 Introduction

Over the years, computational chemistry suites have significantly accelerated scientific research and hold great potential for addressing current and future scientific challenges.[1] In particular, quantum chemistry (QC) software such as GAMESS,[2,3] has been crucial in delivering highly accurate calculations based on quantum mechanics theory. Even in the age of artificial intelligence (AI), QC software remains indispensable. Although AI is increasingly prevalent in the scientific domain, the need for high-quality big data, which QC software can provide, continues to grow.

However, it has been recognized that traditional scientific software development practices are facing significant challenges. In a recent article, Di Felice et al.[1] highlighted that "the sustainability and interoperability of computational chemistry software development are among the most pressing issues."

This paper emphasizes the paradigm shift in software development as a strategy to address these challenges. Traditionally, quantum chemistry (QC) software development has been considered a complex and demanding task. A few decades ago, every mature QC package included several sophisticated software components, such as electron-repulsion integral packages, exchange-correlation functional libraries, dense and sparse linear algebra utilities, and numerical optimization tools. Today, most of the functionalities required to develop new QC software are available in numerous third-party open-source libraries such as BLAS,[4] LAPACK,[5] libint,[6] libXC,[7] CCpy,[8] PCMsolver,[9] libECP,[10] libecpint,[11] libEFP,[12] APBS,[13] and others. The proliferation of these independently distributed developments have significantly reduced the effort required to develop new quantum chemical software by minimizing redundant code development. Moreover, the adoption of interpreter environments like Python marks a departure from traditional executable programs, as demonstrated by projects like eChem.[14] This approach facilitates flexible execution and integration of code, promoting easy prototyping and operation in environments such as Jupyter notebooks and the cloud. Together, these trends suggest a shift towards a "collaborative open-source ecosystem," in contrast to the traditional all-in-one approach that has been the norm in quantum chemical software development.

A crucial factor for the success of this new paradigm is the development of a flexible platform, where modularized quantum chemistry code remains autonomous while allowing for easy interconnection with third-party modules without requiring substantial code modifications.

In this paper, we propose the Open Quantum Platform (**OpenQP**) as a realization of the collaborative open-source ecosystem. The source code of **OpenQP** is already available.[15] The recently developed Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory (MRSF-TDDFT; MRSF for brevity)[16,17] has been utilized as a model theory to exemplify this philosophy. MRSF is a versatile theory that integrates both static multi-configurational effects and dynamic electron correlation into the description of ground and excited electronic states within the computationally efficient linear response theory,[16,18–24] thereby offering promising tools for addressing challenging chemical problems. In the following sections, the basic structures and concepts of **OpenQP** and its corresponding Python wrapper **PyOQP** will be introduced, followed by an overview of their scientific features.

2 Structure of Open Quantum Platform (OpenQP)

As illustrated in Fig. 1, the Open Quantum Platform (**OpenQP**) is built on several layers of components:

- **Fortran base library**: This layer consists of fundamental primitive routines and third-party libraries that form the foundation for higher-level quantum chemical calculations.
- **Quantum Chemical Fortran library** (referred to as **modules** in this paper): These are the applied blocks of quantum chemistry algorithms designed for real-world calculations.
- **Interface**: C API routines that facilitate data exchange and allow third-party software, including Python, to call the corresponding modules.
- **PyOQP**: A reference quantum chemistry program written in Python that utilizes the **module** library (liboqp.so).

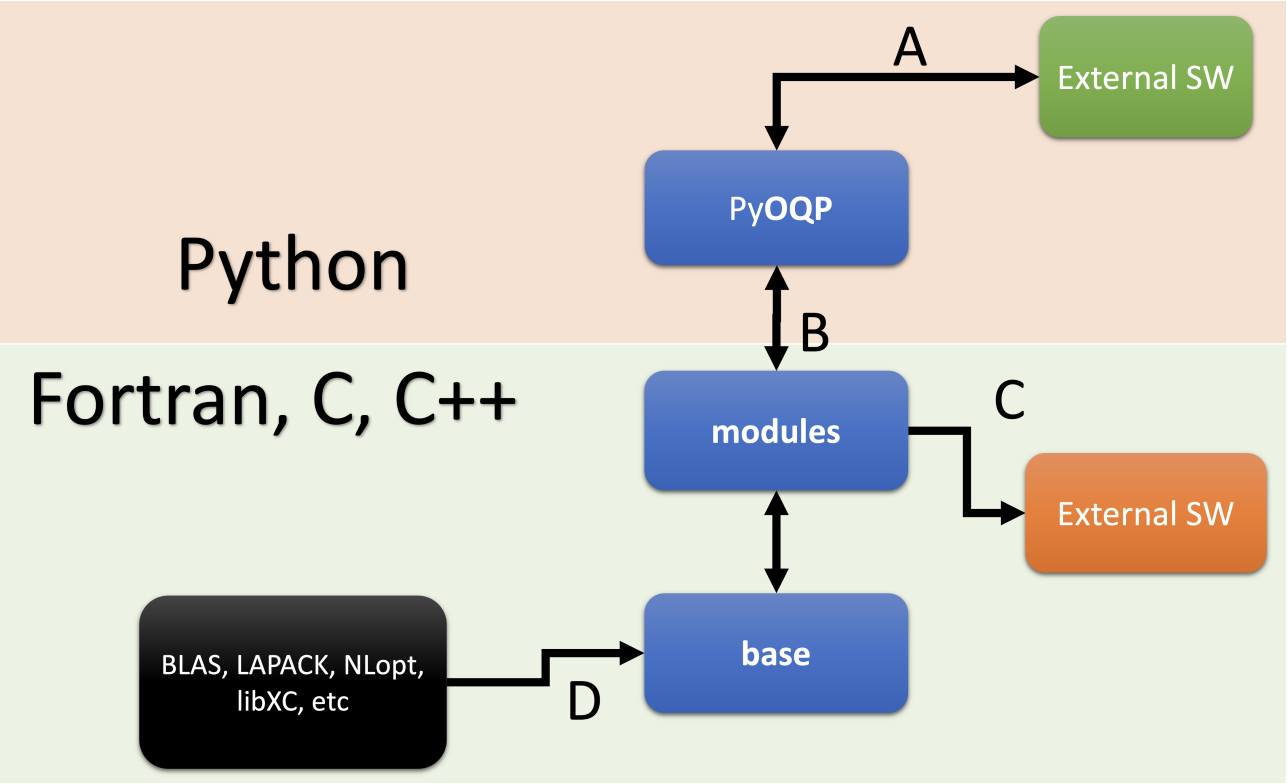


Figure 1: This diagram shows the structural layout of the **OpenQP** framework, highlighting the connections and interactions between different components, such as **base**, **modules**, and external libraries.

2.1 Fortran base library

Table 1: The Fortran Base Library components used by **modules**.

Fortran Source Files	Descriptions
boys, boys_lut, int2, int_rys, rys, rys_lut, int_libint, libint_f, int_rotaxis, int2_pairs, gauss_hermit	Routines for two-electron integrals based on Rys Polynomials, libint by the Valeev Group, and the Rotated Axis method.
grd1, grd2, grd2_rys	Routines for one- and two-electron gradient integrals.
gridint, fuzzycell, molgrid, gridint_tdxc_grad, partfunc, dftgrid, gridint_gxc, gridint_fxc, gridint_grad, gridint_energy, lebedev	Energy and gradient routines for numerical grid integrals in DFT and response theories.
scflib, scf_converger	Main SCF drivers and associated routines.
modules	C API interface.
mathlib, mathlib_types, oqp_linalg, blas_wrap, lapack_wrap, eigen	Basic math libraries and their wrappers.

The core libraries in **OpenQP** consist of essential operations, including one- and two-electron integrals, DFT-related routines, and I/O functionalities, which are utilized by the **modules**. Some representative Fortran source files are listed in Table 1. **OpenQP** integrates several external libraries, such as BLAS,[4] LAPACK,[5] libint,[6] NLopt[25] and libXC,[7] to provide core functionalities for the **modules**. All matrix operations and eigenvalue problems within **OpenQP** rely heavily on BLAS and LAPACK. Non-linear optimizations, such as DIIS during SCF procedures, are handled by NLopt.[25]

Libint[6] offers efficient computation of two-electron repulsion integrals. LibXC[7] is a library of exchange-correlation functionals for density-functional theory (DFT) calculations, supporting local density approximation (LDA) functionals, generalized gradient approximation (GGA) functionals, meta-GGA functionals, and hybrid functionals, by utilizing an interface similar to that developed for GAMESS.[26]

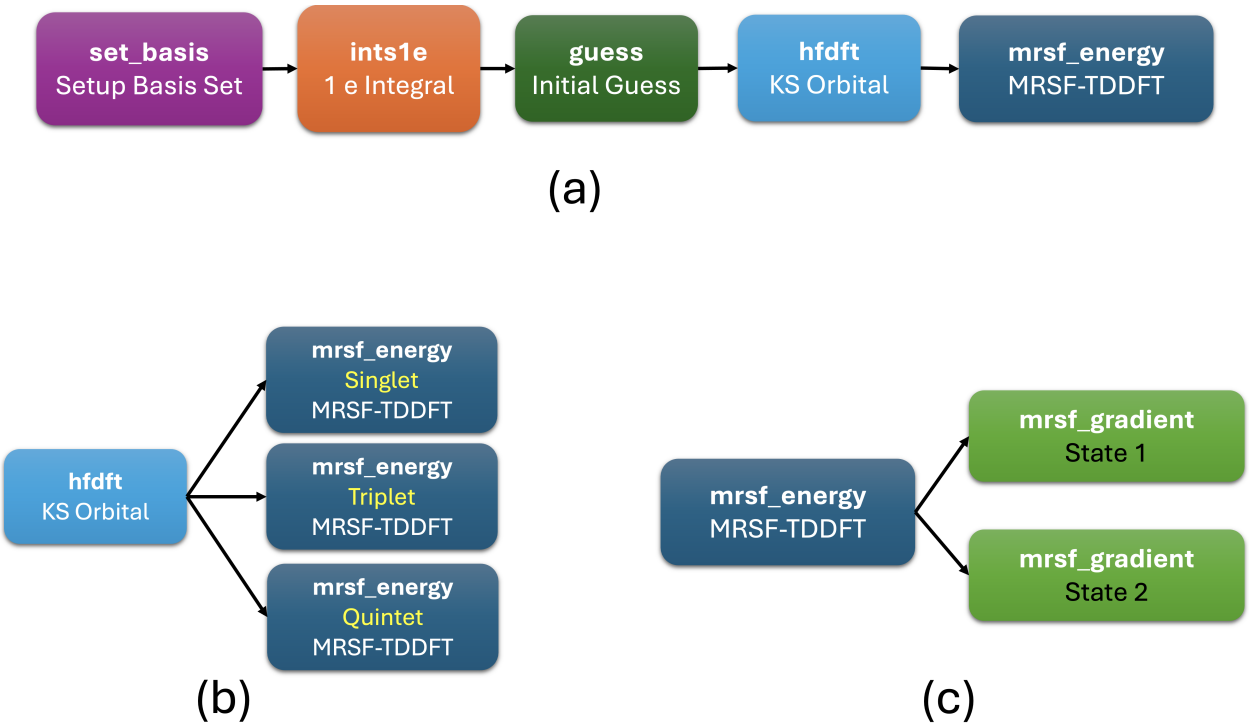


Figure 2: Workflows for (a) an MRSF-TDDFT calculation, (b) branching into three different MRSF-TDDFT calculations with varying spin states derived from the same KS orbitals, and (c) branching into gradient calculations for two different electronic states based on the same MRSF-TDDFT energy calculations, all utilizing Python-accessible **modules**.

2.2 Quantum Chemical Fortran Library (The modules)

Building on the **OpenQP** base library, specialized quantum chemistry functions, referred to as **modules** in this paper, have been developed. These modules include energy and gradient calculations for HF, DFT, TDDFT, SF-TDDFT, and MRSF-TDDFT theories, along with their associated routines. Detailed descriptions of these representative **modules** can be found in the **OpenQP** manuals.[15] They are directly accessible through scripting languages, allowing for seamless integration with third-party modules and software with minimal customization.

To showcase the flexibility of workflow creation using **modules**, three workflows are illustrated in Fig. 2, along with a pseudo-Python code for MRSF-TDDFT calculations in Code 1, utilizing these **modules**. The output from the first module, **oqp.set_basis**, is directly passed to the second module via the structured data **mol**. Five consecutive calls—**set_basis** → **ints_1e** → **guess** → **hfdft** → **MRSF-TDDFT**—complete an MRSF-TDDFT energy calculation, as shown in Fig. 2(a) and Code 1. Each **module** functions independently within the Python environment, offering benefits such as ease of extension, enhancement, and maintenance. For example, if a more advanced initial guess module becomes available, it can simply replace the **oqp.guess** call. Similarly, the MRSF-TDDFT energy calculation using the **oqp.mrsf_energy** module requires only the converged orbitals from the **oqp.hf_dft_energy** module. These two modules are otherwise independent and can reside in different external packages or software, provided data exchange is ensured. Fig. 2(b) and (c) illustrate how workflow branching can be achieved using the same preceding **modules**. The former computes three different spin states using MRSF-TDDFT independently from the same KS orbitals, while the latter computes the gradients of two different electronic states from the same MRSF-TDDFT energy calculations.

```
import oqp
...
# Setting Up Basis Set Information, where mol is the structured data.
oqp.set_basis(mol)
# Perform 1 Electron Integrals.
oqp.ints_1e(mol)
```

```
# Perform Initial Guess.
oqp.guess(mol)
# Perform DFT Energy Calculation.
oqp.hfdft(mol)
# Perform MRSF-TDDFT Energy Calculation.
oqp.mrsf_energy(mol)
```

Code 1: A Python script utilizing four **modules** of **OpenQP** that accomplishes MRSF-TDDFT calculation.

2.3 Interface

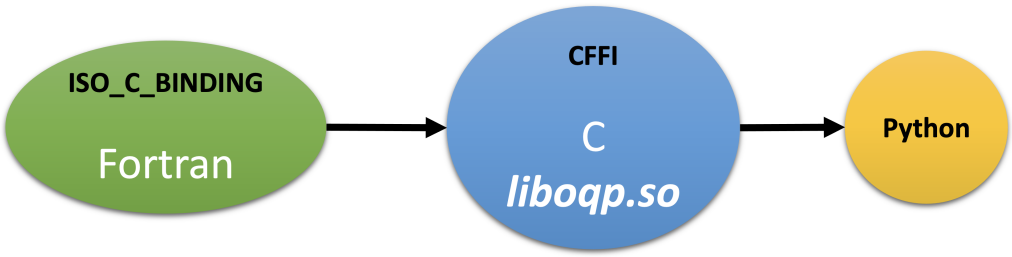


Figure 3: It illustrates the interoperability between various layers and modules within the Open Quantum Platform (**OpenQP**). The core is built using high-performance compiled languages like Fortran and C/C++, which are interfaced with Python to offer flexibility and ease of use.

The modular workflow approach, as demonstrated in the previous section, is implemented through the interfacial layer of **OpenQP**. Traditionally, quantum chemistry software relies on compiled languages like Fortran and C/C++ to maximize hardware performance. In contrast, interpreted languages such as Python offer dynamic typing and high-level data structures, allowing developers to write less code for complex tasks, though typically at the cost of performance.

To harness the strengths of both language types, the core routines of the Open Quantum Platform (**OpenQP**) are encapsulated in a Python library, **liboqp.so**, using **ISO_C_BINDING**[27] and **CFFI**[28] (see Fig. 3). In this framework, high-performance languages like Fortran, C, and C++ are employed for the underlying library, while Python acts as an accessible execution environment for these optimized routines. This approach merges the flexibility and user-friendliness of Python

with the efficiency and speed of compiled languages.

2.4 Data structure and TagArray

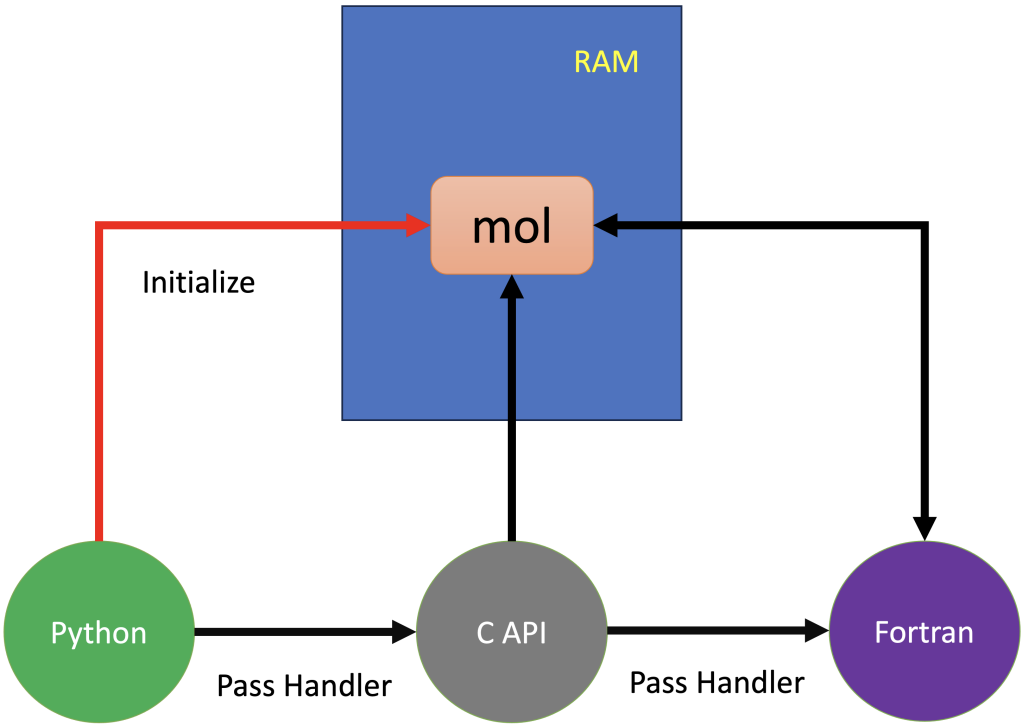


Figure 4: It demonstrates how the common structured data **mol** is shared between the Python and Fortran layers in **OpenQP**. The handler approach reduces data transfer complexities, making it easier to extend workflows.

Combining high-performance compiled languages with the flexible Python environment provides an ideal framework, but it introduces challenges in data sharing across multiple languages. To address this, successful modularization relies on data abstraction—a fundamental concept that hides the complexities of data storage and processing, presenting only essential information in a simplified manner to the user.

To tackle this challenge, we first define a structured data type, **information**, which encompasses all the fundamental *scalar* data needed by **OpenQP**. The Fortran definition of **information** is shown in Code 2, where each element is also a structured data type. This

information is initially allocated in the Python layer as **mol**, as depicted in Fig. 4. Instead of passing individual data through arguments, its handler is passed to Fortran via the C API. By using a single structured data type, **mol**, the complexities of data exchange are minimized, thereby simplifying workflow extension. With this handler, the Fortran layer can directly read from and write to **mol**. Consequently, the calculation results from Fortran modules are saved in **mol**, making them immediately accessible from the Python environment without needing to pass data through a sequence of arguments.

```
! Entire data of OpenQP is composed of a single structured data.
  type, public :: information
    type(molecule) :: mol_prop
    type(energy_results) :: mol_energy
    type(dft_parameters) :: dft
    type(control_parameters) :: control
    type(atomic_structure) :: atoms
    type(functional_t) :: functional
    type(tddft_parameters) :: tddft
    type(container_t) :: dat
    type(basis_set) :: basis
    character(len=:), allocatable :: log_filename
  contains
...
  end type information
```

Code 2: The definition of the structured data type **information** which serves as the primary data structure in **OpenQP**.

Dynamic memory allocations and deallocations are often required during runtime for efficient memory usage, especially when handling large data structures like the Fock matrix. Managing this across different layers can be challenging. To address this, a separate project called "TagArray" has been developed, as illustrated in Fig. 5. TagArray enables any layer within the structure to allocate or deallocate dynamic memory using a Tag, which is a simple text string. Example code in Python and the corresponding Fortran is presented in Code 3, where dynamic memory with

the tag "Python_matrix" is created in Python and accessed from Fortran. This approach not only simplifies the management of dynamic arrays but also facilitates reading from and writing to these arrays across different layers.

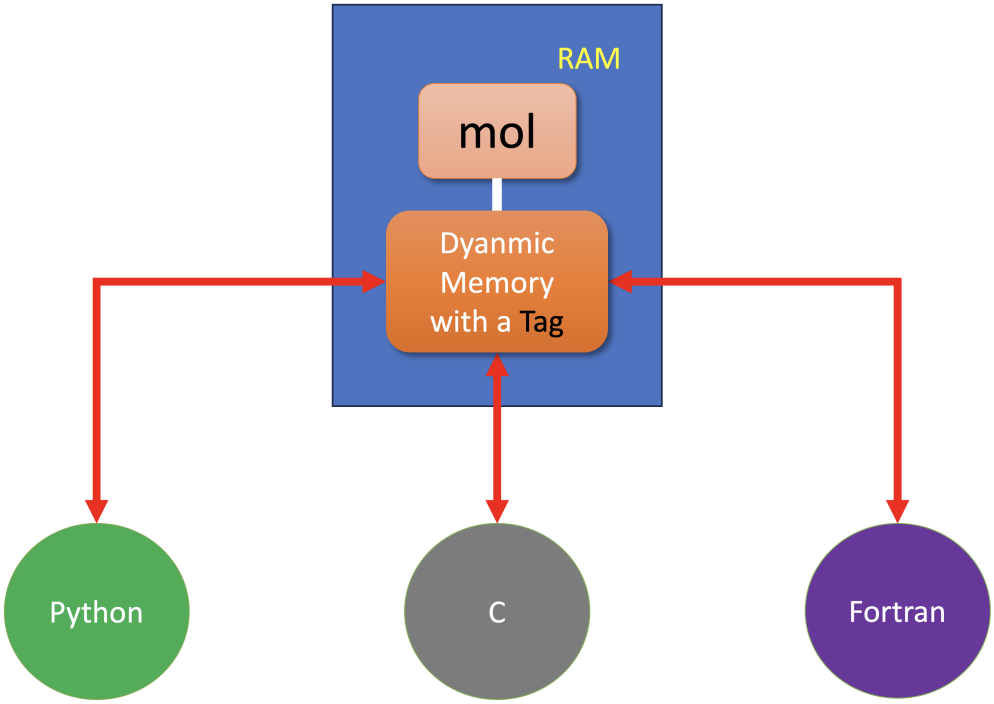


Figure 5: It illustrates how the TagArray project enables dynamic memory allocation and deallocation, allowing any layer of the structure to manage memory efficiently using tags.

```
! A simple Python code to create a Tagaray with a Tag "Python_matrix".
    mol.data["Python_matrix"] = np.reshape([1, 2.0, 3, 4.0, 5, 6], (2,3))
    oqp.Fortran_example(mol)
! A corresponding Fortran code to use the Tagarray defined in Python.
  module example_mod
  implicit none
  character(len=*), parameter :: module_name = "example_mod"
  public Fortran_example
contains
  subroutine Fortran_example(mol)    .....
    use oqp_tagarray_driver, only: data_has_tags, tagarray_get_data, ta_type_real64
   .....
    character(len=*), parameter :: subroutine_name = "Fortran_example"
```

```
! 1) Set the tag of Tagarray for Fortran
  character(len=*), parameter :: tags_general(*) = (/ character(len=80) :: &
      "Python_matrix" /)
! 2) Define a pointer of Python_matrix
  real(kind=dp), contiguous, pointer :: Python_matrix(:,:)
! 3) Load data that was previously allocated from the Python script.
  call data_has_tags(mol%dat, tags_general, module_name, subroutine_name, with_abort)
  call tagarray_get_data(mol%dat, "Python_matrix", Python_matrix)

  do i = 1, ubound(Python_matrix, 1)
    write(*,*) i, (Python_matrix(i,j),j = 1,ubound(Python_matrix, 2))
  end do
 .....
```

Code 3: Python and Fortran code examples for using a TagArray defined in Python.

Since dynamic data can be accessed from the interface layer in Python, existing Python code can be easily modified to incorporate external third-party software or modules, as shown in Fig. 6. For example, dynamic data from TagArray can be passed to third-party software like pyrai2MD,[29] which is also written in Python. This integration enables seamless data sharing and interaction between **OpenQP** and external software, leveraging Python's flexibility and eliminating the need for disk I/O.

2.5 PyOQP: a Python wrapper of OpenQP

In traditional quantum chemistry packages, a significant part of the code is dedicated to tasks unrelated to quantum chemistry, such as input parsing, result printing, and environment setup, leading to redundancy and inefficiency. By leveraging existing third-party tools, like Python runtime environments or specialized software, developers can streamline the process and focus on advancing quantum chemistry methodologies.

In **OpenQP**, only the essential quantum chemical theories are maintained as a library. Its Python wrapper, **PyOQP**, handles file operations, initialization tasks like allocating **mol** data, and

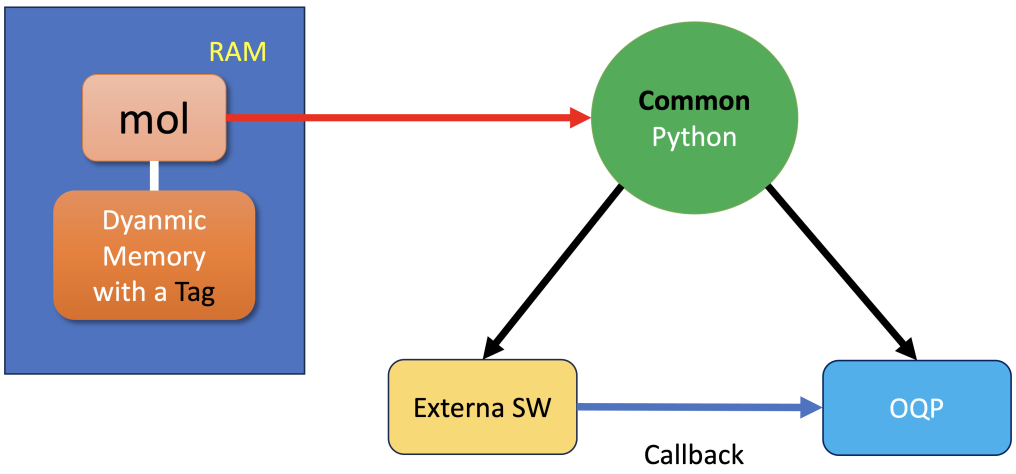


Figure 6: This diagram illustrates the integration of **OpenQP** with third-party software through the Python environment, leveraging dynamic data from Tagarray for seamless data sharing and interaction.

secondary tasks such as geometry optimization and conical intersection searches by incorporating third-party libraries. Currently, **PyOQP** uses external modules like SciPy,[30] libDL-Find,[31] and pyrai2MD[29] to perform basic quantum chemical calculations, including energy, gradient, and Hessian for HF, DFT, TDDFT, SF-TDDFT, and MRSF-TDDFT. SciPy and libDL-Find aid in geometry optimization and conical intersection searches, while pyrai2MD enables nonadiabatic molecular dynamics simulations, to be published separately.

Code 4 demonstrates an example configuration file for **PyOQP**, where input geometries and options for MRSF-TDDFT are pre-set. This script performs a purely quantum mechanical optimization of a C_2H_4 molecule with the PBE0 density functional and a 6-31G basis set. Parsing of keywords and input data is handled by the core routines of Python's **configparser** module. As the in and out file operations, memory management as well as secondary tasks are performed in Python, its modifications and improvements can be rather easily done without revising the **OpenQP** library.

```
# An example file of C2H4
# Energy calculation of MRSF-TDDFT/PBE0/6-31G
# A separate geometry file of c2h4.xyz needs to be provided.
```

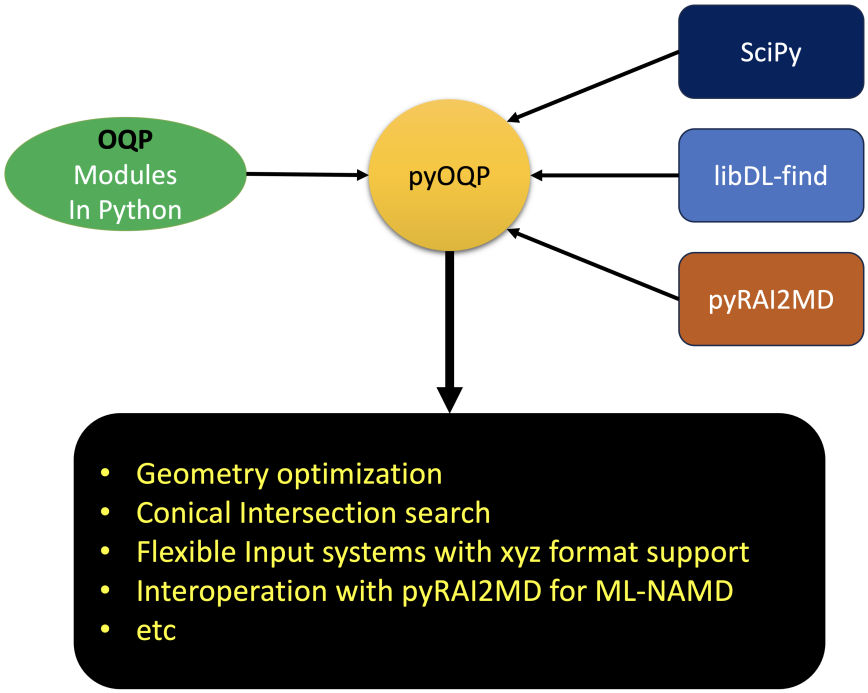


Figure 7: This figure provides an overview of the Python wrapper, **PyOQP**, for showcasing how it enables the execution of **modules** and facilitates memory sharing with external software.

```
[input]
system=c2h4.xyz
runtype=energy
functional=pbe0
method=tdhf
basis=6-31g

[scf]
type=rohf
multiplicity=3

[tdhf]
type=mrsf
nstate=5
```

Code 4: An Example Input File of **PyOQP**

3 The Quantum Chemical Methods in the OpenQP

The **OpenQP** allows to perform quantum chemical calculation for the ground and excited electronic states of molecules. For the ground state calculations, the Hartree-Fock (HF) and density functional theory methods (DFT) can be used for computing energies, energy gradients (w.r.t. atomic movements) and one-electron molecular properties. For the DFT method a broad set of functional is available provided by LibXC library.[7] Currently, LDA, GGA, and meta-GGA (tau-based) functionals are supported. Laplacian-based meta-GGA functionals are not currently implemented, but they are planned for future release. Long-range corrected functional such as CAM-B3LYP are also available in **OpenQP**.

For modeling excited states, the **OpenQP** offers a range of linear-response methods, including time-dependent DFT (TDDFT), spin-flip TDDFT (SF-TDDFT), and the recently developed mixed-reference DFT (MRSF-TDDFT). Additionally, the latter two methods also support ground-state calculations. All methods provide energies and analytical gradients for available functionals. For the TDDFT method, both random-phase and Tamm-Dancoff approximations can be used. The summary of the **OpenQP** main features is presented in the Table 2.

Table 2: Features Present in Open Quantum Platform

Feature	Description
Hartree-Fock method	Energy and gradient calculations with RHF, ROHF, UHF (**Ground State**)
Density Functional Theory (DFT)	Energy and gradient calculations with RHF, ROHF, UHF (**Ground State**)
TDDFT	Energy and gradient calculations, including RPA and TDA, based on RHF, ROHF, UHF (**Excited States**)
SF-TDDFT	Energy and gradient calculations with ROHF, UHF (**Ground and Excited States**)
MRSF-TDDFT	Energy and gradient calculations with ROHF (**Ground and Excited States for Singlet, Triplet, and Quintet**)
New XC Functionals	Includes DTCAM-VAEE, DTCAM-XI, DTCAM-XIV, DTCAM-VEE, and DTCAM-AEE via libXC
Nonadiabatic Coupling	TLF(0), TLF(1), and TLF(2) methods
Parallelization	Supports OpenMP and MPI

3.1 Code Design Principles of Computational Kernels

The primary computational kernels in the methods mentioned above include:

- Calculation of electron-repulsion integrals (ERIs) and their derivatives with respect to nuclear positions.
- Calculation of exchange-correlation (XC) integrals and their various derivatives.

Algorithm 1 "Engine-consumer" design in **OpenQP**. Blue lines are done by the "engine", yellow – by the "consumer"

```
1: Initialization, preliminary calculations
2: Allocate storage for the results
3: loop across basis set shell combinations / molecular grid points
4:     Compute a set of data (ERIs, XC values, etc.)
5:     Process the data and store the result
6: end loop
7: Finalize results
8: Clean up
```

Both the ERI and XC kernels follow a common calculation scheme, summarized in Algorithm 1. The main steps include: (1) a preparation phase, (2) a loop involving data generation (ERI, XC values) and immediate usage, and (3) a cleanup phase. While the kernels share many tasks, only a few are specific to each. To reduce programming effort and enhance flexibility in developing new methods, we employ a design that separates the logic of intermediate data generation and consumption into two entities: an "engine" and a "consumer." The "engine" (a function or non-polymorphic Fortran type) handles tasks common to both ERI and XC kernels. It computes intermediate data (e.g., ERIs, XC functional values at grid points) in chunks and stores them in scratch memory with a specific format. The "consumer," a polymorphic Fortran type, carries out method-specific tasks, such as contracting ERIs with density-like matrices, updating Kohn-Sham matrices, and computing gradient contributions. The "consumer" also manages the reduction of calculation results during parallel runs. By selecting sufficiently large data chunks, the overhead of calling polymorphic type-bound methods can be minimized.

Whenever feasible, we utilize BLAS level 3 operations in the XC kernels. These operations include computing density and its derivatives at grid points (`DGEMM`, `DSYMM`) and updating Kohn-Sham and related matrices (`DSYRK`, `DSYR2K`).

3.1.1 ERI calculation

The two-electron repulsion integrals (ERI) are six-dimensional integrals defined as:

$$(ij|kl) = \iint_{3D\times 3D} \phi_i(\mathbf{r}_1)\phi_j(\mathbf{r}_1)\frac{1}{r_{12}}\phi_k(\mathbf{r}_2)\phi_l(\mathbf{r}_2)d\mathbf{r}_1 d\mathbf{r}_2 \tag{1}$$

where ϕ_i, ϕ_j, ϕ_k, and ϕ_l are atomic orbitals, and r_{12} is the inter-electronic distance. Several algorithms of ERI evaluations are known.[32–38] **OpenQP** utilizes a combination of the rotated-axis based MacMurchie-Davidson[39] and the Rys quadrature algorithm[40] for ERI calculations. These methods are efficient for evaluating two-electron integrals for low and high angular momentum, respectively. In addition, **OpenQP** can be optionally compiled with Libint[6] library support.

3.1.2 Numerical grids for DFT

The most important kernel in the DFT and related methods is the integration of an exchange-correlation functional over the 3D space. For example, the exchange-correlation energy E_{xc} is defined as:

$$E_{\text{xc}}[\rho] = \int \rho(\mathbf{r})\varepsilon_{\text{xc}}(\rho(\mathbf{r}), \nabla\rho(\mathbf{r}), ...)d\mathbf{r} \tag{2}$$

where $\rho(\mathbf{r})$ is the electron density, and ε_{xc} is the exchange-correlation energy per particle. These integrals are typically solved numerically by discretizing the space into a grid and approximating the integral as a sum over grid points:

$$E_{\text{xc}} \approx \sum_i w_i\, \rho(\mathbf{r}_i)\varepsilon_{\text{xc}}(\rho(\mathbf{r}_i), \nabla\rho(\mathbf{r}_i), ...), \tag{3}$$

where $\mathbf{r}_i$ and w_i represent the coordinates of the grid points and their associated weights, respectively. Various schemes for grid setup and weight determination include uniform grids,

Table 3: DFT Features Present in Open Quantum Platform

Features	Details
Radial grids	Murray-Handy-Laming, Mura-Knowles, Treutler-Ahlrichs, Becke
Angular grids	Lebedev
Grid partitioning	Stratmann, Becke, Erf [NOTE: unpublished NWChem], Smoothstep polynomials (2-5 orders)
XC functionals	LDA, GGA, metaGGA (only τ-based) and hybrid functionals in libXC

radial grids for atoms, and adaptive grids.

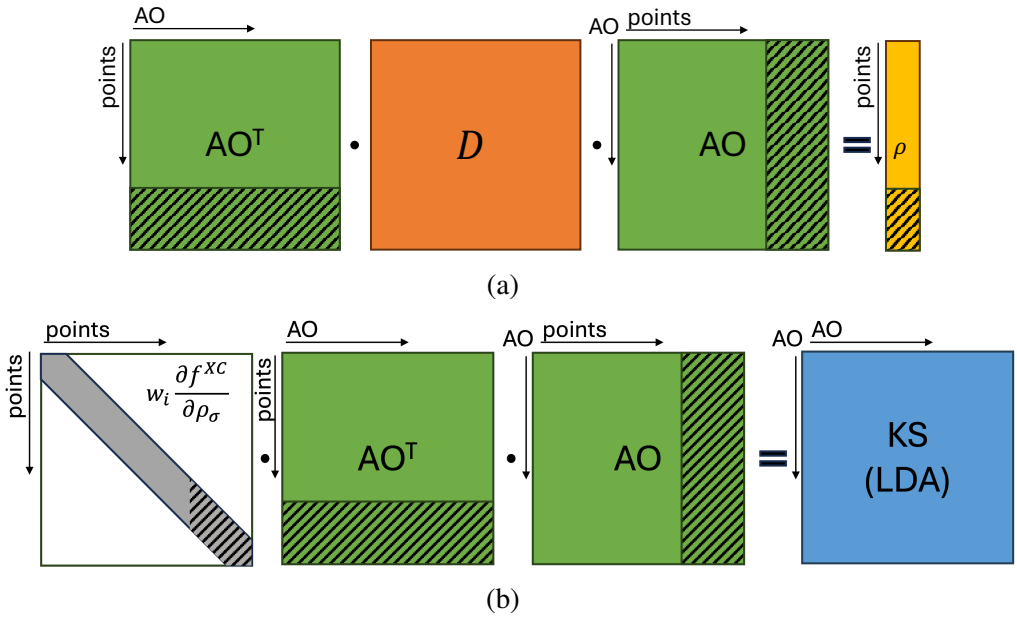


Figure 8: Grid pruning explained: (a) computing the electronic density at grid points, and (b) computing the XC contribution to the Kohn-Sham matrix using the LDA functional as an example. In (a), AO represents the atomic orbital values at grid points, D denotes the density matrix, KS represents the Kohn-Sham matrix, and ρ indicates the electronic density at grid points. In panel (b), the values of the XC functional at grid points ($\partial f^{\mathrm{XC}}/\partial \rho_\sigma$) are multiplied by the grid point weights (w_i), with the result represented as a diagonal matrix. The dashed lines indicate matrix/vector elements that are discarded due to the pruning procedure.

In the current implementation of **OpenQP**, the XC integration grid for a molecule is assembled from ball-shaped atomic grids following Becke's 'fuzzy cell' procedure[41] and a variant proposed by Stratmann et al.[42] The atomic grids are constructed as a direct product of radial and angular grids, with Lebedev angular grids used in **OpenQP** for their high accuracy on a sphere. Several radial grid types are available in **OpenQP**, as listed in Table 3, including

the Murray-Handy-Laming (MHL) grid,[43] Mura-Knowles Log3 grid,[44] Treutler-Ahlrichs (TA) grid,[45] and Becke's grid.[46] For grid re-weighting, a broad set of partitioning functions is available, including the function proposed by Stratmann, Scuseria, and Frisch (SSF),[42] Becke's function,[41] the unpublished erf-2 scheme from the NWchem program, and smooth-step polynomials of orders 2 to 5.

The number of grid points in quantum chemistry calculations is typically very large, often involving several thousand points per atom. To reduce computational effort, grid pruning is employed, where points with negligible contributions to the final results are removed. In **OpenQP**, a dynamic pruning scheme is used: points with small contributions to the integrals (e.g., those with low grid weight or negligible density) are discarded at runtime. Alternatively, specially designed grids with less dense angular coverage in regions close to and far from atomic nuclei can be utilized. Several pruned grid sets, like SG-1,[47] are available. Grid pruning significantly reduces the number of calculations required to compute functional values and their contributions to the Kohn-Sham matrix, as illustrated in Figure 8.

Another important technique is atomic orbital (AO) screening (see Fig. 9). To reduce computational load during the Kohn-Sham matrix update, AOs with negligible density at a given grid point are excluded from the calculations. A list of active orbitals is dynamically constructed at runtime for each batch of grid points. Contributions to the Kohn-Sham matrix are first accumulated in a temporary memory buffer, which is then used to update the matrix. Both grid pruning and AO screening are essential for achieving linear scaling in computational quantum chemistry.

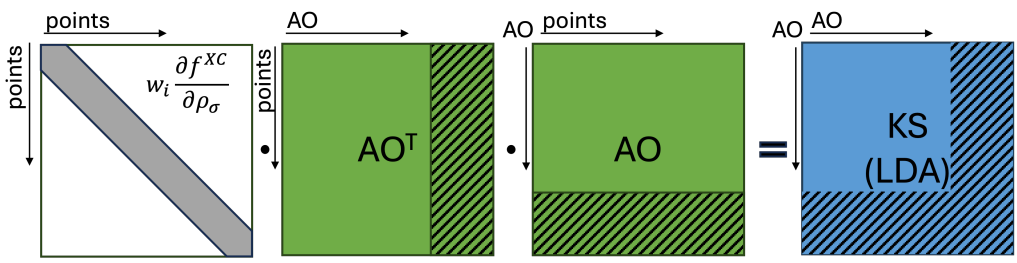


Figure 9: AO screening explained: computing the XC contribution to the Kohn-Sham matrix using the LDA functional as an example. For notations, please refer to the caption of Figure 8.

3.2 Linear Response Theories with the Emphasis on MRSF-TDDFT

One of the main features of **OpenQP** is MRSF-TDDFT[16,17] and its related properties, which offer multiple advantages compared to DFT, TDDFT, and SF-TDDFT:

- **Eliminating Spin Contamination in SF-TDDFT**: MRSF-TDDFT employs a Mixed-Reference Reduced Density Matrix (MR-RDM) to generate two completely independent linear-response equations for singlet and triplet excited states. This approach effectively eliminates spin contamination and enables the automatic identification of spin states, which has been a major challenge for SF-TDDFT.

- **Expanded Response Space with Doubly Excited Configurations**: The expended response space by MR-RDM naturally incorporates additional nondynamic electron correlations, thereby providing a more balanced description of response states. It also increases the number of doubly excited configurations, which are essential for accurately describing excited states and recovering the missing states in TDDFT.

- **Ground Singlet State (S_0)**: Unlike TDDFT, both SF-TDDFT and MRSF-TDDFT can represent the ground singlet (S_0) state as one of their response states. MRSF-TDDFT further improves upon this by eliminating spin contamination.

- **Description of Open-Shell Singlets**: Therefore, MRSF-TDDFT can effectively describe open-shell singlets, such as diradicals,[20] and accurately model bond-breaking processes, addressing a significant limitation of DFT.[22]

- **Accurate Topology of Conical Intersections**: MRSF-TDDFT's capability to generate the ground singlet state (S_0) resolves the topological conical intersection issues encountered in TDDFT as well as all single reference theories,[24] as it represents both S_0 and S_1 within the same response states.

- **Core-Hole Relaxation**: The high-spin triplet reference in MRSF-TDDFT provides a

straightforward method for ensuring core-hole relaxation, leading to accurate results in X-ray Absorption Spectroscopy (XAS).[48]

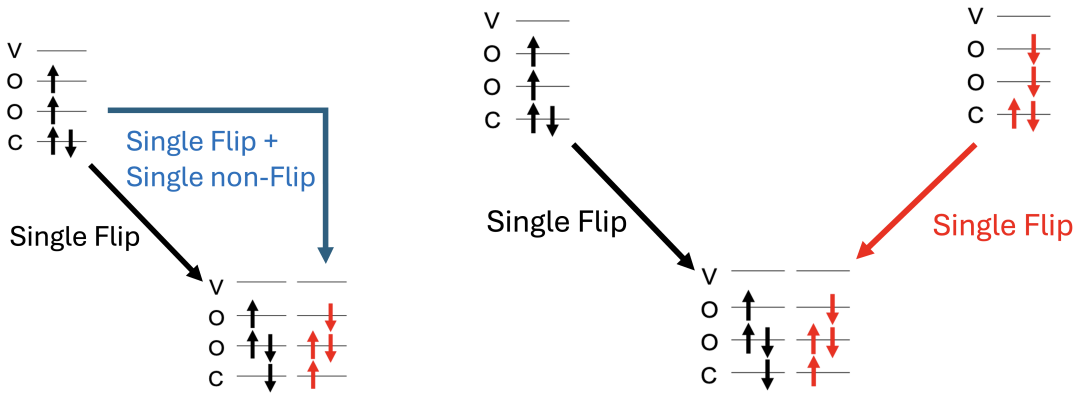
Linear-Response (LR)-TDDFT[49–51] is based on the time-dependent Kohn–Sham (TD–KS) equation with the linear-response formalism using a closed-shell singlet ground state as its reference. LR-TDDFT has become the most popular method for studying valence-excited states due to its computational simplicity and efficiency.[51] Given a common set of reference orbitals, typically obtained by Kohn-Sham DFT, LR-TDDFT can conveniently produce a full spectrum of singly excited states in a single calculation without prior knowledge of the nature of the states. This capability allows easy access to various properties, including excitation energies of all excited states, and inter-state properties such as transition dipole moments (TDM), non-adiabatic coupling (NAC), spin-orbit coupling (SOC), and core excitations.

Despite its advantages, there are well-known limitations of LR-TDDFT, such as in describing long-range charge transfer excitations,[52–56] doubly excited states,[57–59] bond breaking,[60,61] and real and avoided conical intersections (CI).[62–65] Some drawbacks of TDDFT, particularly the incorrect description of conical intersections and the poor handling of multi-reference electronic states, can be addressed by spin-flip (SF)-TDDFT.[66–68] However, the use of an open-shell high-spin triplet reference ($M_S = +1$) in SF-TDDFT introduces significant spin contamination. A fundamental solution to this issue is to recover the missing configurations by incorporating higher excitations, as illustrated in Fig. 10(a). Nevertheless, this approach entails considerable computational overheads.

Instead of introducing high-rank excitations from a single reference, a second reference ($M_S = -1$) was introduced (see Fig. 10(b)) as an alternative way of expanding the response space by some of us,[16] yielding MRSF-TDDFT.[16,17] In the realization of this idea, a new mixed reference state was introduced that has an equiensemble density of the $M_S = +1$ and $M_S = -1$ components of a triplet state.

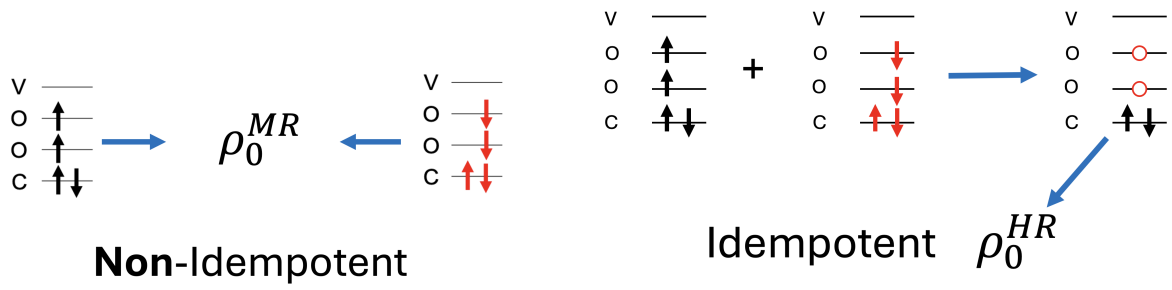
$$\rho_0^{\mathrm{MR}}(x) = \frac{1}{2}\left\{\rho_0^{M_S=+1}(x) + \rho_0^{M_S=-1}(x)\right\}. \tag{4}$$

However, a simple addition of the two densities makes the resulting density ρ_0^{MR}



(a) (b)

Figure 10: (a) The black configuration is obtained through a single spin-flip excitation from the ROHF reference with $M_S = +1$, while the red configuration requires a combination of both spin-flip and non-spin-flip excitations. (b) The red configuration can also be derived from the red ROHF reference with $M_S = -1$ by a single spin-flip excitation.



(a) (b)

Figure 11: (a) The combination of the two ROHF references with $M_S = +1$ and $M_S = -1$ results in a non-idempotent density. (b) Introducing new spin functions, represented by red circles, enables us to combine the two ROHF references into a single hypothetical reference, yielding an idempotent density.

non-idempotent (see Fig. 11(a)). To address this challenge, a spin function transformation was introduced, which transforms the α and β spin functions into new spin functions, σ_+ and σ_-. These two new spin functions allow us to represent the two references of $M_S = +1$ and $M_S = -1$ with a single hypothetical reference and the corresponding density ρ_0^{MR}, as shown in Fig. 11(b).

$$\sigma_+ = \frac{(1+i)\alpha + (1-i)\beta}{2} \tag{5a}$$

$$\sigma_- = \frac{(1-i)\alpha + (1+i)\beta}{2}, \tag{5b}$$

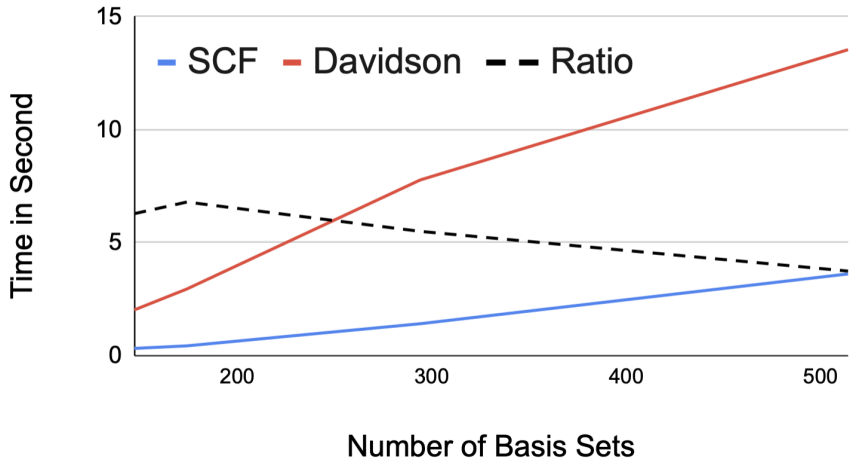


Figure 12: Wall clock timings per cycle for the SCF (blue) and response (Davidson iteration) parts (red) as a function of the number of basis functions in MRSF-TDDFT calculations. The ratio of Davidson to SCF timings is also shown with a dashed line. All calculations were performed using the new quantum code in an OpenMP parallel environment with 16 cores of Intel(R) Xeon(R) CPU E5-2699A v4 @ 2.40GHz.

MRSF-TDDFT is expressly designed to furnish a balanced treatment of dynamic and nondynamic electron correlations for both ground and excited states—an essential feature given the sensitivity of torsional potential energy surfaces to the quality of quantum mechanical theories. Comprehensive benchmarks have attested to the promise of MRSF-TDDFT as a viable alternative in addressing the challenges posed by quantum chemical studies.[18–24,69–77] Notably,

the judicious treatment of electron correlation in MRSF-TDDFT facilitates reliable NAMD simulations, as demonstrated in a series of recent studies.[23,78–83] A recent perspective further underscores the merits of MRSF-TDDFT in this context.[84]

A simple benchmark is presented in Fig. 12, where the timings per iteration for the SCF and response parts (Davidson) are compared. The Davidson/SCF ratio (dashed line) decreases as the number of basis functions increases. Generally, the SCF part requires more iterations than the response part in MRSF-TDDFT, resulting in a similar overall overhead for both. This demonstrates the efficiency of the current **OpenQP** code for response calculations.

3.3 New Exchange-Correlation Functionals for MRSF-TDDFT

Despite the advantages, it is also evident that the performance of MRSF-TDDFT is subjective to the choice of exchange-correlation (XC) functionals due to its density functional origin.

To address this issue, our recent work has focused on optimizing exchange-correlation (XC) functionals for MRSF-TDDFT[86] based on the "double tuning" approach, as shown in Fig. 13. This idea has achieved remarkable reductions (by a factor of two) in mean absolute errors (MAEs) across all XC functionals (see bottom of Fig. 13). The "double tuning" approach within the framework of the Coulomb attenuated method (CAM) functional led to the development of two new XC functionals: doubly tuned Coulomb attenuated method (DTCAM)-VEE and DTCAM-AEE, for vertical excitation energies (VEEs).[86] Additionally, the concept of "valence attenuation," where the amount of exact exchange for the frontier orbital regions is selectively reduced, was also introduced, resulting in improved functionals such as DTCAM-VAEE.[87] These new functionals are summarized in Table 4 along with their performances in Table 5. The latter table shows the MAEs (Mean Absolute Energies) of the new functionals.

For core Ionization Potentials (cIPs), we utilized the CORE65 set,[88] which includes 65 1s excitation energies from 32 small molecules. For valence Ionization Potentials (vIPs), we referred to the benchmark data provided by Sherrill and co-workers,[89] which consists of 21 experimental values for 24 medium-sized organic molecules. Additionally, Thiel's set[85] and Loos's set[90] were

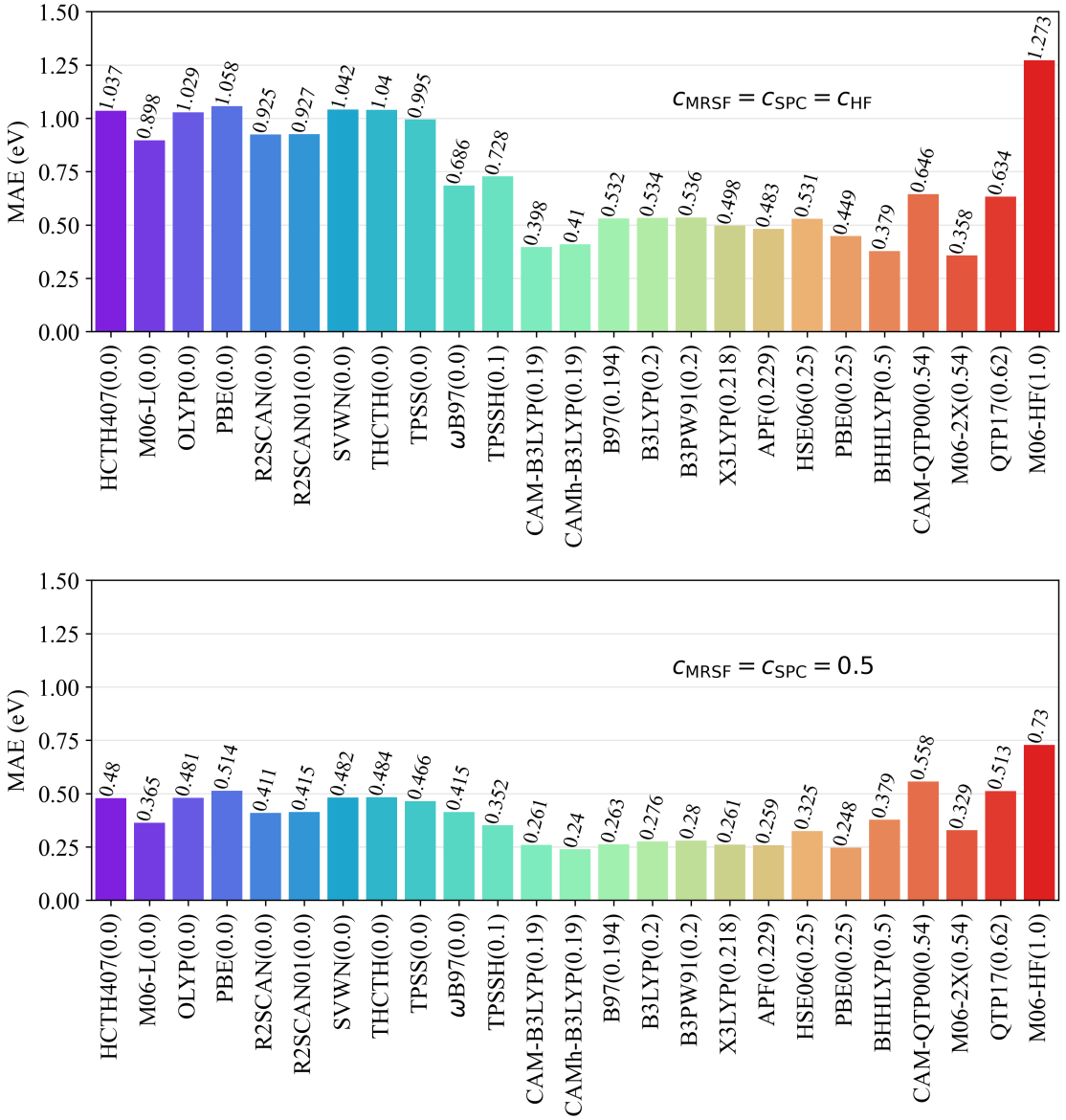


Figure 13: The Mean Absolute Errors (MAEs) (in eV) for the S1 and S2 excited states, categorized by symmetry label, compared with TBE-2 for Thiel's molecular set as a function of XC functionals.[85] The top figure presents the MAE results with cMRSF=cSPC=c_{HF}, while the bottom figure shows results with c_{MRSF}=c_{SPC}=0.5. All calculations were performed using MRSF-TDDFT with the 6-31G(d) basis set. The values in parentheses represent c_{HF} for each functional. It is important to note that the actual percentage of c_{HF} in ωB97, CAM-B3LYP, and CAMh-B3LYP ranges from 0.0 to 1.0, 0.19 to 0.65, and 0.19 to 0.50, respectively. The collinear formulation of the current MRSF-TDDFT method eliminates contributions from pure XC functionals such as VWN5 and LYP. Figures taken from Ref. 86.

Table 4: New Exchange-Correlation Functionals for MRSF-TDDFT

Functional	Description
DTCAM-VEE	Doubly tuned XC functional for Vertical Excitation Energy (VEE) based on Thiel's benchmark set.
DTCAM-AEE	Doubly tuned XC functional for VEE, optimized based on Thiel's set, conical intersections (CIs) in *trans*-butadiene and thymine, and nonadiabatic molecular dynamics (NAMD) simulations on thymine.
DTCAM-XI	XC functional specifically designed for core ionization potentials (cIPs) and valence ionization potentials (vIPs).
DTCAM-VAEE	Optimized for VEEs using both the "double tuning" and "valence attenuation" concepts.
DTCAM-XIV	XC functional aimed at consistently predicting both VEEs and ionization potentials (IPs) simultaneously, based on the "double tuning" and "valence attenuation" concepts.

employed to evaluate Vertical Excitation Energies (VEEs).

Table 5: MAEs of various XC functionals against three benchmark sets of cIP, VEE and vIP. The data are taken from Ref. 87

Functional	cIP	VEE	vIP	Average
QTP17	0.90	0.91	0.49	0.77
B3LYP	15.7	0.60	2.55	6.28
BH&HLYP	2.73	0.68	1.25	1.55
CAM-B3LYP	13.6	0.64	1.00	5.08
DTCAM-VEE	14.6	0.56	1.34	5.50
DTCAM-AEE	14.6	0.59	0.90	5.36
DTCAM-XI	**0.51**	0.97	**0.30**	0.59
DTCAM-XIV	0.64	0.67	0.61	0.64
DTCAM-VAEE	3.47	**0.57**	1.41	1.82

3.4 Numerical Computations of Nonadiabatic Coupling

A component along the x direction of the first-order nonadiabatic coupling (h_{IJ}^x, vibronic coupling) and derivative coupling (d_{IJ}^x) vectors (DCV) are defined as follows:

$$h_{IJ}^x = \langle \Psi_I | \frac{\partial H}{\partial x} | \Psi_J \rangle, \tag{6}$$

$$d_{IJ}^x = \langle \Psi_I | \frac{\partial}{\partial x} | \Psi_J \rangle = \frac{h_{IJ}^x}{E_J - E_I}, \tag{7}$$

between the adiabatic states $\langle \Psi_I |$ and $|\Psi_J\rangle$. These vectors, together with the energy gradient difference vector (GDV), define the branching plane of conical intersections (CIs),[91–95] as well as the rate of the nonadiabatic population transfer at and near conical intersections and avoided crossings between the states I and J. Given that only a limited number of quantum chemical computational methods are capable of accurately describing the topology of CIs,[62,64,96–98] computational studies of DCV in scientific literature remain relatively rare.

Within the finite difference approximation, DCV in Eq. (7) is obtained from the overlap integrals between the (*non-orthogonal*) states *I* and *J* calculated at the geometries displaced by Δx as

$$d_{IJ}^x = \frac{1}{2\Delta x} \left(\langle \Psi_I(x - \frac{1}{2}\Delta x) | \Psi_J(x + \frac{1}{2}\Delta x) \rangle - \langle \Psi_I(x + \frac{1}{2}\Delta x) | \Psi_J(x - \frac{1}{2}\Delta x) \rangle \right) + O(\Delta x^2). \tag{8}$$

Although Eq. (8) has a simple form, evaluation of the overlap integrals (OIs) by the Löwdin formula[99] requires significant computation effort. For example, obtaining the OI in connection with the single-excitation methods, such as TDDFT and SF-TDDFT, typically requires a computational effort approximately scaling as $O(n_{\text{occ}}^5 n_{\text{vir}}^2)$ with the number of occupied (n_{occ}) and virtual (n_{vir}) molecular orbitals (MOs).[70,100] Based on Wick's theorem, a new method scaling approximately as $O(n^4)$ was recently proposed by Burton.[101]

To address these challenges, we proposed a series of fast and accurate specialized algorithms for overlap integral (OI) evaluation based on the truncated Leibniz formula (TLF).[70] A major

advantage of the TLF methods is their negligible computational overhead with quadratic scaling ($O(n^2)$) in the case of TLF(0). The TLF(1) and TLF(2) approximations include first- and second-order contributions, respectively. All three approximations have been implemented in **OpenQP**, enabling routine computations of derivative coupling vectors (DCV) and non-adiabatic couplings (NAC). An example of DCV values calculated by MRSF-TDDFT, along with previous results, is summarized in Table 6, showing excellent agreement with MRCISD results.[96]

Table 6: Normalized DCVs of twisted-pyramidalized MECI (S_1/S_0) of $CH_2NH_2^+$ at the MRSDCI optimized geometries. The DCV of MRCISD is taken from Ref. 96.

Atom	MRCISD			MRSF-TDDFT		
	x	y	z	x	y	z
N	-0.001713	-0.006346	-0.002218	0.000923	-0.006521	-0.002291
C	0.003539	0.024967	0.008541	-0.000168	0.024196	0.008289
H	0.053196	0.141633	-0.449465	0.055744	0.148537	-0.467819
H	0.002468	-0.490929	-0.170891	0.002729	-0.474192	-0.164545
H	-0.003467	0.498836	0.173529	-0.002829	0.481910	0.167163
H	-0.054151	-0.167982	0.440453	-0.055575	-0.173343	0.458263

3.5 Parallel Execution by OpenMP and MPI

The parallel execution of **OpenQP** is currently achieved through a combination of OpenMP (Open Multi-Processing) and MPI (Message Passing Interface). OpenMP supports multi-platform shared memory multiprocessing programming. In contrast, MPI is a standardized and portable message-passing system designed for parallel computing architectures with distributed memory. Together, these tools facilitate parallel programming on platforms ranging from simple desktop machines to large-scale symmetric multiprocessors.

OpenMP utilizes a set of compiler directives, library routines, and environment variables that influence run-time behavior, while MPI provides a framework for communication between processes running on different nodes in a distributed memory environment, enabling efficient data exchange and synchronization across multiple processors.

As a result, the combination of OpenMP and MPI can harness optimal parallel efficiencies

across both shared and distributed memory architectures, making it a powerful approach for maximizing performance in diverse computational environments.

4 Perspectives

Future developments include the implementation of techniques such as EKT-MRSF-TDDFT[71] and SOC-MRSF-TDDFT.[102]

The analysis of excited states using linear response methodologies often relies on molecular orbitals (MOs) of the reference state or other orbitals, such as natural or natural transition orbitals. However, these orbitals lack corresponding energies, hindering the development of a consistent orbital picture for excited states. EKT-MRSF-TDDFT addresses this issue by producing Dyson orbitals and their associated orbital energies (electron binding energies) for response states, providing a clear and chemically meaningful representation of electronic transitions, and bridging the gap between orbital theory and response computations.

EKT can be formulated as a generalized eigenproblem:[103]

$$\tilde{\mathbf{W}}^K \, \mathbf{C}^K = \mathbf{D}^K \, \mathbf{C}^K \, \boldsymbol{\varepsilon}^K \tag{9}$$

where $\mathbf{D}^K$ is the relaxed density matrix,[104] used when the energy is not obtained variationally, and $\tilde{\mathbf{W}}^K$ is the Lagrangian matrix of the initial electronic state K (e.g., a neutral molecule in state K).[105,106] The vectors $\mathbf{C}^K$ contain the expansion coefficients of the Dyson orbitals $\varphi_k^K(\mathbf{r})$ in terms of the eigenvectors (natural orbitals) of the density matrix $\mathbf{D}^K$. As the Lagrangian matrix imposes orbital orthonormality constraints, the derivative of these constraints occurs in the energy gradient obtained with any method where orbital orthogonality is preserved.[107] Thus, the Dyson orbitals and their respective electron binding energies can be obtained with any computational method for which the analytical energy gradient is available.[105,106]

Relativistic MRSF-TDDFT was developed by considering the spin-orbit coupling (SOC) within the mean-field approximation. The Hamiltonian can be separated into a spin-independent

part ($\hat{H}_0$) and a spin-dependent part ($\hat{H}_{\mathrm{SOC}}$):

$$\hat{H} = \hat{H}_0 + \hat{H}_{\mathrm{SOC}} \tag{10}$$

The spin-independent term includes the conventional non-relativistic Hamiltonian and the scalar relativistic Hamiltonian, while the spin-dependent term contains the spin-orbit coupling (SOC) operator. The SOC effect is introduced in a perturbative manner. Accordingly, the spin-independent Hamiltonian ($\hat{H}_0$) is solved using the standard MRSF-TDDFT response calculations.[16,17] The resulting SOC-MRSF-TDDFT faithfully reproduces experimental results with high accuracy[77] making it a promising electronic structure protocol for challenging situations, such as nonadiabatic molecular dynamics (NAMD) that incorporate both internal conversions and intersystem crossings in large systems.

The current limitations of ROKS will also be addressed by developing UKS (Unrestricted Kohn-Sham) based MRSF-TDDFT (UMRSF-TDDFT),[108] which will undoubtedly expand its scope of applicability.

Many features, such as effective core potentials and the polarizable continuum model (PCM), still need to be implemented. Community support is highly encouraged and welcomed.

5 Conclusion

In this paper, we introduced the Open Quantum Platform (**OpenQP**), a new quantum chemistry platform specifically designed to tackle the challenges of sustainability and interoperability in computational chemistry.

OpenQP advocates for a "collaborative open-source ecosystem", a departure from the traditional centralized, all-in-one approach commonly seen in quantum chemical software development. **OpenQP** addresses this need by offering a suite of popular quantum chemical theories as autonomous **modules**, including energy and gradient calculations for HF, DFT,

TDDFT, SF-TDDFT, and MRSF-TDDFT. These **modules** are designed for easy integration with third-party software and can be directly accessed through the Python wrapper **PyOQP**. Together, **OpenQP** and **PyOQP** combine high-performance compiled modules with the flexibility of Python, enabling straightforward prototyping and operation in adaptable environments like Jupyter notebooks.

A key scientific innovation in **OpenQP** is the Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory (MRSF-TDDFT) and its custom exchange-correlation functionals, such as the DTCAM series (VAEE, XI, XIV, AEE, and VEE). MRSF-TDDFT effectively combines static multi-configurational effects with dynamic electron correlation, offering a balanced and computationally efficient method for studying ground and excited electronic states. The tailored exchange-correlation functionals for MRSF-TDDFT further expand its applicability. Additionally, the platform's implementation of efficient parallel execution using OpenMP and MPI, along with advanced libraries like BLAS and LAPACK, ensures that **OpenQP** delivers high performance and accuracy in quantum chemical calculations.

Looking ahead, future developments for **OpenQP** include the integration of advanced techniques such as EKT-MRSF-TDDFT and SOC-MRSF-TDDFT, which will enhance the platform's ability to handle properties of molecular orbital origin and relativistic effects. These advancements highlight **OpenQP**'s potential to remain at the cutting edge of quantum chemical research.

In short, **OpenQP** sets a new standard in quantum chemical software development by combining modularity, interoperability, and the innovative quantum chemical theory of MRSF-TDDFT. Its advanced features and planned future enhancements position it as a critical tool for advancing our understanding of quantum mechanical phenomena and meeting the challenges of future software development.

Acknowledgments

This work was supported by the NRF funded by the Ministry of Science and ICT (2020R1A2C2008246 and 2020R1A5A1019141).

References

(1) Di Felice, R.; Mayes, M. L.; Richard, R. M.; Williams-Young, D. B.; Chan, G. K.-L.; de Jong, W. A.; Govind, N.; Head-Gordon, M.; Hermes, M. R.; Kowalski, K., et al. A perspective on sustainable computational chemistry software development and integration. *Journal of Chemical Theory and Computation* **2023**, *19*, 7056–7076.

(2) Barca, G. M. J. et al. Recent developments in the general atomic and molecular electronic structure system. *The Journal of Chemical Physics* **2020**, *152*, 154102.

(3) Zahariev, F. et al. The General Atomic and Molecular Electronic Structure System (GAMESS): Novel Methods on Novel Architectures. *Journal of Chemical Theory and Computation* **2023**, *19*, 7031–7055, PMID: 37793073.

(4) Blackford, L. S.; Petitet, A.; Pozo, R.; Remington, K.; Whaley, R. C.; Demmel, J.; Dongarra, J.; Duff, I.; Hammarling, S.; Henry, G., et al. An updated set of basic linear algebra subprograms (BLAS). *ACM Transactions on Mathematical Software* **2002**, *28*, 135–151.

(5) Anderson, E.; Bai, Z.; Bischof, C.; Blackford, S.; Demmel, J.; Dongarra, J.; Du Croz, J.; Greenbaum, A.; Hammarling, S.; McKenney, A.; Sorensen, D. *LAPACK Users' Guide*, 3rd ed.; Society for Industrial and Applied Mathematics: Philadelphia, PA, 1999.

(6) Valeev, E. F. Libint: A library for the evaluation of molecular integrals of many-body operators over Gaussian functions. http://libint.valeyev.net/, 2022; version 2.8.0.

(7) Lehtola, S.; Steigemann, C.; Oliveira, M. J.; Marques, M. A. Recent developments in libxc—A comprehensive library of functionals for density functional theory. *SoftwareX* **2018**, *7*, 1–5.

(8) Group, P. R. CCpy: A coupled-cluster package written in Python. 2024; `https://piecuch-group.github.io/ccpy/`.

(9) Remigio, R. D.; Steindal, A. H.; Mozgawa, K.; Weijo, V.; Cao, H.; Frediani, L. PCMSolver: An Open-Source Library for Solvation Modeling. *International Journal of Quantum Chemistry* **2018**, *119*.

(10) Developers, L. libECP: Library for Effective Core Potentials. 2021; `https://github.com/chrr/libECP`, Accessed: 2024-07-15.

(11) Shaw, J. G.; Hill, J. Effective core potentials for high accuracy quantum chemistry. *Journal of Chemical Physics* **2017**, *147*, 074108.

(12) Gordon, M. S. et al. LibEFP: A new parallel implementation of the effective fragment potential method. *Journal of Chemical Theory and Computation* **2017**, *13*, 4032–4040.

(13) Jurrus, E. et al. Improvements to the APBS biomolecular solvation software suite. *Journal of Chemical Physics* **2018**, *27*, 112–128.

(14) Fransson, T.; Delcey, M. G.; Brumboiu, I. E.; Hodecker, M.; Li, X.; Rinkevicius, Z.; Dreuw, A.; Rhee, Y. M.; Norman, P. eChem: A Notebook Exploration of Quantum Chemistry. *Journal of Chemical Education* **2023**, *100*, 1664–1671.

(15) OpenQP: Open Quantum Platform. `https://github.com/Open-Quantum-Platform/openqp`.

(16) Lee, S.; Filatov, M.; Lee, S.; Choi, C. H. Eliminating spin-contamination of spin-flip time dependent density functional theory within linear response formalism by the use of

zeroth-order mixed-reference (MR) reduced density matrix. *J. Chem. Phys.* **2018**, *149*, 104101.

(17) Lee, S.; Kim, E. E.; Nakata, H.; Lee, S.; Choi, C. H. Efficient implementations of analytic energy gradient for mixed-reference spin-flip time-dependent density functional theory (MRSF-TDDFT). *J. Chem. Phys.* **2019**, *150*, 184111.

(18) Horbatenko, Y.; Lee, S.; Filatov, M.; Choi, C. H. Performance Analysis and Optimization of Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory (MRSF-TDDFT) for Vertical Excitation Energies and Singlet–Triplet Energy Gaps. *J. Phys. Chem. A* **2019**, *123*, 7991.

(19) Horbatenko, Y.; Lee, S.; Filatov, M.; Choi, C. H. How Beneficial Is the Explicit Account of Doubly-Excited Configurations in Linear Response Theory? *J. Chem. Theory Comput.* **2021**, *17*, 975–984.

(20) Horbatenko, Y.; Sadiq, S.; Lee, S.; Filatov, M.; Choi, C. H. Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory (MRSF-TDDFT) as a Simple yet Accurate Method for Diradicals and Diradicaloids. *J. Chem. Theory Comput.* **2021**, *17*, 848–859.

(21) Baek, Y. S.; Lee, S.; Filatov, M.; Choi, C. H. Optimization of Three State Conical Intersections by Adaptive Penalty Function Algorithm in Connection with the Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory Method (MRSF-TDDFT). *J. Phys. Chem. A* **2021**, *125*, 1994–2006.

(22) Lee, S.; Horbatenko, Y.; Filatov, M.; Choi, C. H. Fast and Accurate Computation of Nonadiabatic Coupling Matrix Elements Using the Truncated Leibniz Formula and Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory. *J. Phys. Chem. Lett.* **2021**, *12*, 4722–4728.

(23) Park, W.; Lee, S.; Huix-Rotllant, M.; Filatov, M.; Choi, C. H. Impact of the Dynamic

Electron Correlation on the Unusually Long Excited-State Lifetime of Thymine. *J. Phys. Chem. Lett.* **2021**, *12*, 4339–4346.

(24) Lee, S.; Shostak, S.; Filatov, M.; Choi, C. H. Conical Intersections in Organic Molecules: Benchmarking Mixed-Reference Spin-Flip Time-Dependent DFT (MRSF-TD-DFT) vs Spin-Flip TD-DFT. *J. Phys. Chem. A* **2019**, *123*, 6455–6462.

(25) Johnson, S. G. The NLopt nonlinear-optimization package. 2014; `http://github.com/stevengj/nlopt`.

(26) Gerasimov, I. S.; Zahariev, F.; Leang, S. S.; Tesliuk, A.; Gordon, M. S.; Medvedev, M. G. Introducing LibXC into GAMESS (US). *Mendeleev Communications* **2021**, *31*, 302–305.

(27) International Organization for Standardization, Information technology — Programming languages — Fortran — Part 1: Base language. 2004.

(28) Rigo, A.; Fijalkowski, M. CFFI - Foreign Function Interface for Python calling C code. `https://cffi.readthedocs.io`, 2013; `https://cffi.readthedocs.io`, Python package for interfacing with C code.

(29) Li, J.; Reiser, P.; Boswell, B. R.; Eberhard, A.; Burns, N. Z.; Friederich, P.; Lopez, S. A. Automatic discovery of photoisomerization mechanisms with nanosecond machine learning photodynamics simulations. *Chemical Science* **2021**, *12*, 5302–5314.

(30) Virtanen, P. et al. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods* **2020**, *17*, 261–272.

(31) Kästner, J.; Carr, J. M.; Keal, T. W.; Thiel, W.; Wander, A.; Sherwood, P. DL-FIND: An Open-Source Geometry Optimizer for Atomistic Simulations. *J. Phys. Chem. A* **2009**, *113*, 11856–11865.

(32) Boys, S. Electronic wave functions. I. A general method of calculation for the stationary

states of any molecular system. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* **1950**, *200*, 542–554.

(33) Obara, S.; Saika, A. Efficient recursive computation of molecular integrals over Cartesian Gaussian functions. *The Journal of Chemical Physics* **1986**, *84*, 3963–3974.

(34) Pople, J. A.; Beveridge, D. L.; Dobosh, P. A. Approximate self-consistent molecular orbital theory. V. Intermediate neglect of differential overlap. *The Journal of Chemical Physics* **1967**, *47*, 2026–2032.

(35) Head-Gordon, M.; Pople, J. A.; Frisch, M. J. A method for two-electron Gaussian integral and integral derivative evaluation using recurrence relations. *The Journal of Chemical Physics* **1988**, *89*, 5777–5786.

(36) Gill, P. M.; Johnson, B. G.; Pople, J. A.; Frisch, M. J. Analytic evaluation of integrals over Gaussian basis functions. *International Journal of Quantum Chemistry* **1991**, *40*, 745–758.

(37) McMurchie, L. E.; Davidson, E. R. One- and two-electron integrals over Cartesian Gaussian functions. *Journal of Computational Physics* **1978**, *26*, 218–231.

(38) Schlegel, H. B. The transformation of two-electron integrals: A comparative study. *Theoretical Chemistry Accounts* **1989**, *75*, 143–156.

(39) Ishimura, K.; Nagase, S. A new algorithm of two-electron repulsion integral calculations: a combination of Pople–Hehre and McMurchie–Davidson methods. *Theoretical Chemistry Accounts* **2008**, *120*, 185–189.

(40) Dupuis, M.; Rys, J.; King, H. F. Evaluation of molecular integrals over Gaussian basis functions. *The Journal of Chemical Physics* **1986**, *65*, 111–116.

(41) Becke, A. D. A multicenter numerical integration scheme for polyatomic molecules. *The Journal of Chemical Physics* **1988**, *88*, 2547–2553.

(42) Stratmann, R. E.; Scuseria, G. E.; Frisch, M. J. Achieving linear scaling in exchange-correlation density functional quadratures. *Chemical physics letters* **1996**, *257*, 213–223.

(43) Murray, C. W.; Handy, N. C.; Laming, G. J. Quadrature schemes for integrals of density functional theory. *Molecular Physics* **1993**, *78*, 997–1014.

(44) Mura, M. E.; Knowles, P. J. Improved radial grids for quadrature in quantum mechanical calculations. *The Journal of Chemical Physics* **1996**, *104*, 9848–9858.

(45) Treutler, O.; Ahlrichs, R. Efficient molecular numerical integration schemes. *The Journal of Chemical Physics* **1995**, *102*, 346–354.

(46) Becke, A. D. A multicenter numerical integration scheme for polyatomic molecules. *The Journal of Chemical Physics* **1988**, *88*, 2547–2553.

(47) Gill, P. M. W.; Johnson, B. G.; Pople, J. A.; Frisch, M. J. Standard Grid 1: A new grid for accurate molecular quadrature. *Chemical Physics Letters* **1992**, *197*, 499–505.

(48) Park, W.; Alías-Rodríguez, M.; Cho, D.; Lee, S.; Huix-Rotllant, M.; Choi, C. H. Mixed-reference spin-flip time-dependent density functional theory for accurate x-ray absorption spectroscopy. *Journal of Chemical Theory and Computation* **2022**, *18*, 6240–6250.

(49) Runge, E.; Gross, E. K. Density-functional theory for time-dependent systems. *Physical review letters* **1984**, *52*, 997.

(50) Casida, M. E.; Chong, D. Recent advances in density functional methods. *Computational Chemistry: Reviews of Current Trends* **1995**,

(51) Casida, M. E.; Huix-Rotllant, M. Progress in time-dependent density-functional theory. *Annual review of physical chemistry* **2012**, *63*, 287–323.

(52) Dreuw, A.; Weisman, J. L.; Head-Gordon, M. Long-range charge-transfer excited states in time-dependent density functional theory require non-local exchange. *The Journal of chemical physics* **2003**, *119*, 2943–2946.

(53) Dreuw, A.; Head-Gordon, M. Single-reference ab initio methods for the calculation of excited states of large molecules. *Chemical reviews* **2005**, *105*, 4009–4037.

(54) Dev, P.; Agrawal, S.; English, N. J. Determining the appropriate exchange-correlation functional for time-dependent density functional theory studies of charge-transfer excitations in organic dyes. *The Journal of Chemical Physics* **2012**, *136*, 224301.

(55) Maitra, N. T. Undoing static correlation: Long-range charge transfer in time-dependent density-functional theory. *The Journal of chemical physics* **2005**, *122*, 234104.

(56) Baerends, E.; Gritsenko, O.; Van Meer, R. The Kohn–Sham gap, the fundamental gap and the optical gap: the physical meaning of occupied and virtual Kohn–Sham orbital energies. *Physical Chemistry Chemical Physics* **2013**, *15*, 16408–16425.

(57) Cave, R. J.; Zhang, F.; Maitra, N. T.; Burke, K. A dressed TDDFT treatment of the 2^1A_g states of butadiene and hexatriene. *Chemical Physics Letters* **2004**, *389*, 39–42.

(58) Neugebauer, J.; Baerends, E. J.; Nooijen, M. Vibronic coupling and double excitations in linear response time-dependent density functional calculations: Dipole-allowed states of N_2. *The Journal of chemical physics* **2004**, *121*, 6155–6166.

(59) Maitra, N. T.; Zhang, F.; Cave, R. J.; Burke, K. Double excitations within time-dependent density functional theory linear response. *The Journal of Chemical Physics* **2004**, *120*, 5932–5937.

(60) Aryasetiawan, F.; Gunnarsson, O.; Rubio, A. Excitation energies from time-dependent density-functional formalism for small systems. *EPL (Europhysics Letters)* **2002**, *57*, 683.

(61) Filatov, M. Ensemble DFT approach to excited states of strongly correlated molecular systems. *Density-functional methods for excited states* **2015**, 97–124.

(62) Levine, B. G.; Ko, C.; Quenneville, J.; Martínez, T. J. Conical intersections and double excitations in time-dependent density functional theory. *Molecular Physics* **2006**, *104*, 1039–1051.

(63) Huix-Rotllant, M.; Filatov, M.; Gozem, S.; Schapiro, I.; Olivucci, M.; Ferré, N. Assessment of density functional theory for describing the correlation effects on the ground and excited state potential energy surfaces of a retinal chromophore model. *Journal of chemical theory and computation* **2013**, *9*, 3917–3932.

(64) Gozem, S.; Melaccio, F.; Valentini, A.; Filatov, M.; Huix-Rotllant, M.; Ferré, N.; Frutos, L. M.; Angeli, C.; Krylov, A. I.; Granovsky, A. A., et al. Shape of multireference, equation-of-motion coupled-cluster, and density functional theory potential energy surfaces at a conical intersection. *Journal of chemical theory and computation* **2014**, *10*, 3074–3084.

(65) Ferré, N.; Filatov, M.; Huix-Rotllant, M.; Adamo, C. *Density-functional methods for excited states*; Springer, 2016; Vol. 368.

(66) Li, Z.; Liu, W. Theoretical and numerical assessments of spin-flip time-dependent density functional theory. *The Journal of chemical physics* **2012**, *136*, 024107.

(67) Wang, F.; Ziegler, T. Time-dependent density functional theory based on a noncollinear formulation of the exchange-correlation potential. *The Journal of chemical physics* **2004**, *121*, 12191–12196.

(68) Shao, Y.; Head-Gordon, M.; Krylov, A. I. The spin–flip approach within time-dependent density functional theory: Theory and applications to diradicals. *The Journal of chemical physics* **2003**, *118*, 4807–4818.

(69) Park, W.; Shen, J.; Lee, S.; Piecuch, P.; Filatov, M.; Choi, C. H. Internal Conversion between Bright ($1^1B_u^+$) and Dark ($2^1A_g^-$) States in s-trans-Butadiene and s-trans-Hexatriene. *J. Phys. Chem. Lett.* **2021**, *12*, 9720–9729.

(70) Lee, S.; Kim, E.; Lee, S.; Choi, C. H. Fast Overlap Evaluations for Nonadiabatic Molecular Dynamics Simulations: Applications to SF-TDDFT and TDDFT. *J. Chem. Theory Comput.* **2019**, *15*, 882.

(71) Pomogaev, V.; Lee, S.; Shaik, S.; Filatov, M.; Choi, C. H. Exploring Dyson's Orbitals and Their Electron Binding Energies for Conceptualizing Excited States from Response Methodology. *J. Phys. Chem. Lett.* **2021**, *12*, 9963–9972.

(72) Kim, H.; Park, W.; Kim, Y.; Filatov, M.; Choi, C. H.; Lee, D. Relief of excited-state antiaromaticity enables the smallest red emitter. *Nat. Commun.* **2021**, *12*, 1–9.

(73) Lee, S.; Park, W.; Nakata, H.; Filatov, M.; Choi, C. H. Recent advances in ensemble density functional theory and linear response theory for strong correlation. *Bull. Korean Chem. Soc.* **2022**, *43*, 17–34.

(74) Japahuge, A.; Lee, S.; Choi, C. H.; Zeng, T. Design of singlet fission chromophores with cyclic (alkyl)(amino) carbene building blocks. *J. Chem. Phys.* **2019**, *150*, 234306.

(75) Pradhan, E.; Lee, S.; Choi, C. H.; Zeng, T. Diboron-and diaza-doped anthracenes and phenanthrenes: their electronic structures for being singlet fission chromophores. *J. Phys. Chem. A* **2020**, *124*, 8159–8172.

(76) James, D.; Pradhan, E.; Lee, S.; Choi, C. H.; Zeng, T. Dicarbonyl anthracenes and phenanthrenes as singlet fission chromophores. *Can. J. Chem.* **2022**, *99*, 1–10.

(77) Komarov, K.; Park, W.; Lee, S.; Zeng, T.; Choi, C. H. Accurate Spin–Orbit Coupling by Relativistic Mixed-Reference Spin-Flip-TDDFT. *Journal of Chemical Theory and Computation* **2023**,

(78) Park, W.; Shen, J.; Lee, S.; Piecuch, P.; Joo, T.; Filatov, M.; Choi, C. H. Dual Fluorescence of Octatetraene Hints at a Novel Type of Singlet-to-Singlet Thermally Activated Delayed Fluorescence Process. *J. Phys. Chem. C* **2022**,

(79) Park, W.; Filatov, M.; Sadiq, S.; Gerasimov, I.; Lee, S.; Joo, T.; Choi, C. H. A Plausible Mechanism of Uracil Photohydration Involves an Unusual Intermediate. *J. Phys. Chem. Lett.* **2022**, *13*, 7072–7080.

(80) Huix-Rotllant, M.; Schwinn, K.; Pomogaev, V.; Farmani, M.; Ferré, N.; Lee, S.; Choi, C. H. Photochemistry of thymine in solution and DNA revealed by an electrostatic embedding QM/MM combined with mixed-reference spin-flip TDDFT. *Journal of Chemical Theory and Computation* **2022**,

(81) Lee, S.; Park, W.; Nakata, H.; Filatov, M.; Choi, C. H. *Time-Dependent Density Functional Theory*; Jenny Stanford Publishing, 2023; pp 101–139.

(82) Shostak, S.; Park, W.; Oh, J.; Kim, J.; Lee, S.; Nam, H.; Filatov, M.; Kim, D.; Choi, C. H. Ultrafast Excited State Aromatization in Dihydroazulene. *Journal of the American Chemical Society* **2023**,

(83) Sadiq, S.; Park, W.; Mironov, V.; Lee, S.; Filatov, M.; Choi, C. H. Prototropically Controlled Dynamics of Cytosine Photodecay. *The Journal of Physical Chemistry Letters* **2023**, *14*, 791–797.

(84) Park, W.; Komarov, K.; Lee, S.; Choi, C. H. Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory: Multireference Advantages with the Practicality of Linear Response Theory. *J. Phys. Chem. Lett.* **2023**, *14*, 8896–8908.

(85) Schreiber, M.; Silva-Junior, M. R.; Sauer, S. P. A.; Thiel, W. Benchmarks for electronically excited states: CASPT2, CC2, CCSD, and CC3. *Journal of Chemical Physics* **2008**, *128*, 134110.

(86) Komarov, K.; Park, W.; Lee, S.; Huix-Rotllant, M.; Choi, C. H. Doubly Tuned Exchange–Correlation Functionals for Mixed-Reference Spin-Flip Time-Dependent Density Functional Theory. *Journal of Chemical Theory and Computation* **2023**, *19*, 7671–7684.

(87) Park, W.; Lashkaripour, A.; Komarov, K.; Lee, S.; Huix-Rotllant, M.; Choi, C. H. Toward Consistent Predictions of Core/Valence Ionization Potentials and Valence Excitation Energies by MRSF-TDDFT. *Journal of Chemical Theory and Computation* **2024**, *20*, 5679–5694.

(88) Golze, D.; Keller, L.; Rinke, P. Accurate Absolute and Relative Core-Level Binding Energies from GW. *The Journal of Physical Chemistry Letters* **2020**, *11*, 1840–1847, PMID: 32043890.

(89) Richard, R. M.; Marshall, M. S.; Dolgounitcheva, O.; Ortiz, J. V.; Brédas, J.-L.; Marom, N.; Sherrill, C. D. Accurate Ionization Potentials and Electron Affinities of Acceptor Molecules I. Reference Data at the CCSD(T) Complete Basis Set Limit. *Journal of Chemical Theory and Computation* **2016**, *12*, 595–604, PMID: 26731487.

(90) Loos, P.-F.; Lipparini, F.; Boggio-Pasqua, M.; Scemama, A.; Jacquemin, D. A Mountaineering Strategy to Excited States: Highly Accurate Energies and Benchmarks for Medium Sized Molecules. *Journal of Chemical Theory and Computation* **2020**, *16*, 1711–1741, PMID: 31986042.

(91) Teller, E. The Crossing of Potential Surfaces. *J. Phys. Chem.* **1937**, *41*, 109–116.

(92) Herzberg, G.; Longuet-Higgins, H. Intersection of potential energy surfaces in polyatomic molecules. *Discuss. Faraday Soc.* **1963**, *35*, 77–82.

(93) Longuet-Higgins, H. C. The intersection of potential energy surfaces in polyatomic molecules. *Proc. R. Soc. Lond.* **1975**, *344*, 147–156.

(94) Atchity, G. J.; Xantheas, S. S.; Ruedenberg, K. Potential energy surfaces near intersections. *J. Chem. Phys.* **1991**, *95*, 1862–1876.

(95) Manaa, M. R.; Yarkony, D. R. On the intersection of two potential energy surfaces of the same symmetry. Systematic characterization using a Lagrange multiplier constrained procedure. *J. Chem. Phys.* **1993**, *99*, 5251–5256.

(96) Nikiforov, A.; Gamez, J. A.; Thiel, W.; Huix-Rotllant, M.; Filatov, M. Assessment of approximate computational methods for conical intersections and branching plane vectors in organic molecules. *J. Chem. Phys.* **2014**, *141*, 124122.

(97) Tuna, D.; Lefrancois, D.; Wolański, Ł.; Gozem, S.; Schapiro, I.; Andruniów, T.; Dreuw, A.; Olivucci, M. Assessment of Approximate Coupled-Cluster and Algebraic-Diagrammatic-Construction Methods for Ground-and Excited-State Reaction Paths and the Conical-Intersection Seam of a Retinal-Chromophore Model. *J. Chem. Theory Comput.* **2015**, *11*, 5758–5781.

(98) Kjønstad, E. F.; Myhre, R. H.; Martínez, T. J.; Koch, H. Crossing conditions in coupled cluster theory. *J. Chem. Phys.* **2017**, *147*, 164105.

(99) Löwdin, P.-O. Quantum Theory of Many-Particle Systems. I. Physical Interpretations by Means of Density Matrices, Natural SpinOrbitals, and Convergence Problems in the Method of Configurational Interaction. *Phys. Rev.* **1955**, *97*, 1474–1489.

(100) Ryabinkin, I. G.; Nagesh, J.; Izmaylov, A. F. Fast Numerical Evaluation of Time-Derivative Nonadiabatic Couplings for Mixed Quantum-Classical Methods. *J. Phys. Chem. Lett.* **2015**, *6*, 4200–4203.

(101) Burton, H. G. A. Generalized nonorthogonal matrix elements: Unifying Wick's theorem and the Slater–Condon rules. *The Journal of Chemical Physics* **2021**, *154*, 144109.

(102) Konecny, L.; Repisky, M.; Ruud, K.; Komorovsky, S. Relativistic four-component linear damped response TDDFT for electronic absorption and circular dichroism calculations. *J. Chem. Phys.* **2019**, *151*, 194112.

(103) Peters, G.; Wilkinson, J. H. Ax = λ Bx and the Generalized Eigenproblem. *SIAM J. Num. Analysis* **1970**, *7*, 479–492.

(104) Perera, S. A.; Watts, J. D.; Bartlett, R. J. A theoretical study of hyperfine coupling constants. *J. Chem. Phys.* **1994**, *100*, 1425–1434.

(105) Morrison, R. C.; Liu, G. Extended Koopmans' theorem: Approximate ionization energies from MCSCF wave functions. *J. Comput. Chem.* **1992**, *13*, 1004–1010.

(106) Cioslowski, J.; Piskorz, P.; Liu, G. Ionization potentials and electron affinities from the extended Koopmans' theorem applied to energy-derivative density matrices: The EKTMPn and EKTQCISD methods. *J. Chem. Phys.* **1997**, *107*, 6804–6811.

(107) Pulay, P. Ab initio calculation of force constants and equilibrium geometries in polyatomic molecules. *Mol. Phys.* **1969**, *17*, 197–204.

(108) Komarov, K.; Oh, M.; Nakata, H.; Lee, S.; Choi, C. H. UMRSF-TDDFT: Unrestricted Mixed-Reference Spin-Flip-TDDFT. **2024**,

TOC Graphic