`Fiend` – Finite Element Quantum Dynamics

J. Solanpää, E. Räsänen

Laboratory of Physics, Tampere University of Technology, Tampere FI-33101, Finland

Abstract

We present `Fiend` – a simulation package for three-dimensional single-particle time-dependent Schrödinger equation for cylindrically symmetric systems. `Fiend` has been designed for the simulation of electron dynamics under *inhomogeneous* vector potentials such as in nanostructures, but it can also be used to study, e.g., nonlinear light-matter interaction in atoms and linear molecules. The light-matter interaction can be included via the minimal coupling principle in its full rigour, beyond the conventional dipole approximation. The underlying spatial discretization is based on the finite element method (FEM), and time-stepping is provided either via the generalized-α or Crank-Nicolson methods. The software is written in Python 3.6, and it utilizes state-of-the-art linear algebra and FEM backends for performance-critical tasks. `Fiend` comes along with an extensive API documentation, a user guide, simulation examples, and allows for easy installation via `Docker` or the Python Package Index.

Keywords: Time-dependent Schrödinger equation, finite element method, atoms, nanostructures, light-matter interaction, strong field physics

arXiv:1812.05943v1 [physics.comp-ph] 14 Dec 2018

PROGRAM SUMMARY

Program Title: `Fiend`

Journal Reference:

Catalogue identifier:

Licensing provisions: MIT License

Programming language: Python 3.6

Computer: Tested on x86_64 architecture.

Operating system: Tested on Linux and macOS.

RAM: Simulation dependent: from megabytes to hundreds of gigabytes.

Parallelization: MPI-based parallelization. Thread-based parallelization via `BLAS` and `LAPACK` backends.

Classification: 2.2 Spectra, 2.5 Photon Interactions, 4.3 Differential equations, 6.5 Software including Parallel Algorithms

External routines/libraries:

`PETSc`, `SLEPc`, `FEniCS`, `HDF5`

`petsc4py`, `slepc4py`, `mpi4py`, `h5py`, `numpy`, `scipy`, `matplotlib`, `psutil`, `mypy`, `progressbar2`

Nature of problem:

Solution of time independent and time dependent single active electron Schrödinger equations in cylindrically symmetric systems including interactions with spatially inhomogeneous vector potentials.

Solution method:

Finite element discretization of the Schrödinger equation. Time evolution via the generalized-α or Crank-Nicolson methods.

Restrictions:

Cylindrically symmetric single active electron systems.

Unusual features:

Finite element discretization of the equations allowing *inhomogeneous spatial dependence of the vector potential* (e.g., plasmon-enhanced fields). Integration with the FEniCS finite element suite. Installable from the Python Package Index. Pre-installed `Docker` images available.

Additional comments:

The source code is also available at `https://fiend.solanpaa.fi`.

Python package available at `https://pypi.org/project/fiend/`.

Docker images are available at `https://hub.docker.com/r/solanpaa/fiend/`.

Running time:

From minutes to weeks, depending on the simulation.

1. Introduction

Simulation of three-dimensional (3D) single particle quantum mechanics (QM) is still one of the most used computational approaches in the strong field and attosecond communities. In these fields, the nonlinear interaction of the atomic or molecular electron with the driving laser field requires fast and accurate integration of the 3D time dependent Schrödinger equation (TDSE). Recent applications include, e.g., benchmarking of approximate models [1–6], study of high-order harmonic generation [7–9], and the study of photoionization [10–14].

Recently the strong-field and attosecond communities have turned their attention to related phenomena in nanostructures. There the nanostructure geometry and plasmonic effects cause the electromagnetic field to become extremely inhomogeneous [15] which, in turn, causes significant differences to the traditional ultrafast and strong-field phenomena in atoms and molecules. Recent studies include, e.g., nanostructure-enhanced photoionization of gases [16–18] and electron emission from nanostructures such as tips [19–28] and rods [29–33].

Efficient and accurate simulation of ultrafast and strong-field phenomena in nanostructures and nanostructure-enhanced gases requires specialized TDSE-solvers. Most importantly, since the laser electric field has strong spatial inhomogeneity, it is imperative to have position-dependent control of simulation accuracy. In addition, it would be beneficial to have the TDSE-solver directly interface with a solver for the electromagnetic problem.

To this end, we have developed Fiend, a QM simulator based on finite element method (FEM). Fiend provides solvers for the time independent and time-dependent single active electron Schrödinger equations in cylindrically symmetric systems[1] for solving time-independent and time-dependent Schrödinger equations in cylindrically symmetric systems. We have designed Fiend to integrate with the open source FEniCS finite element suite [34–40] allowing for easy description of complicated system geometries. Being FEM-based, Fiend also allows the use of potentials with integrable singularities.

Fiend is written in Python 3.6, but much of the heavy number-crunching is delegated to well-tested and efficient external libraries. The software is modular and easy to extend. We also provide a comprehensive unit- and integration-test suite to ensure reliability. Moreover, we provide straightforward installation methods either via the Python Package Index [41] or as Docker images [42].

This paper is organized as follows. In Sec. 2 we describe the finite element discretization of the Schrödinger equations and the time-propagation schemes. Section 3 focuses on the design and implementation of Fiend. Section 4 provides a few example simulation, and finally, in Sec. 5 we summarize this paper and possible extensions of the presented software suite.

2. Finite element quantum mechanics

2.1. Systems

Fiend is designed for simulating cylindrically symmetric single (active) electron systems interacting with time and (optionally) position dependent vector potentials. The Hamiltonian operator of these systems can be written as

$$\hat{H} = \hat{T}_\rho + \frac{\hat{p}_z^2}{2} + V(\hat{\rho}, \hat{z}) + W(\hat{\rho}, \hat{z}, \hat{p}_\rho, \hat{p}_z, t), \quad (1)$$

where $\hat{\rho}$ is the planar radial coordinate operator, $\hat{z}$ the z-coordinate operator, $\hat{T}_\rho$ the kinetic energy operator with respect to planar radial motion[2], $\hat{p}_z$ the z-component of the momentum operator, V the static potential, and W the light-matter interaction operator.

Fiend readily supports three types of interaction operators W. First, for linearly polarized vector potentials $\mathbf{A} = f(t)\mathbf{e}_z$ in the length gauge the interaction operator is given by

$$W_{\mathrm{LG}} = -\frac{\partial f(t)}{\partial t}\hat{z}, \quad (3)$$

secondly, in the velocity gauge the interaction operator reads

$$W_{\mathrm{VG}} = f(t)\hat{p}_z, \quad (4)$$

---

[1]Although the restriction to cylindrically symmetric systems can be lifted relatively easily if needed.

[2] Note that $\hat{T}_\rho$ is not proportional to the square of the planar radial momentum operator, i.e.,

$$\hat{T}_\rho = -\frac{1}{2}\left(\frac{\partial^2}{\partial_\rho^2} + \frac{1}{2}\frac{\partial}{\partial_\rho}\right)$$
$$\neq \frac{1}{2}\left[-\mathrm{i}\left(\frac{\partial}{\partial_\rho} + \frac{1}{2\rho}\right)\right]^2 = \frac{\hat{p}_\rho^2}{2}. \quad (2)$$

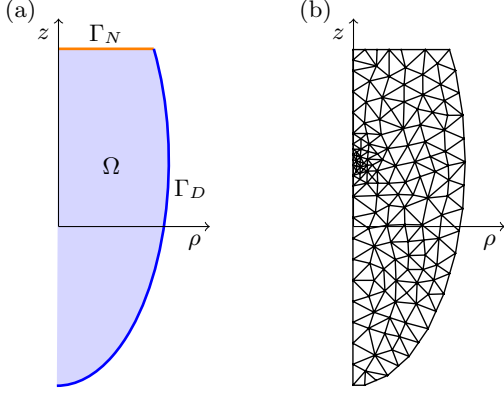This is a peculiarity of quantum mechanics in cylindrical coordinates [43].

Figure 1: (a) Simulation domain Ω is a truncation of the full $\rho \geq 0$ half-plane. We can impose both Dirichlet and Neumann boundary conditions at will on different sections (Γ_D, Γ_N) of the domain boundary $\partial\Omega$. (b) Triangular mesh of the domain Ω supports arbitrary spatial refinement.

and finally for more general cylindrically symmetric inhomogeneous vector potentials $\mathbf{A} = \mathbf{A}_s(\rho, z)f(t)$ we provide the interaction operator

$$W_{\text{inhomogeneous}} = \frac{1}{2}f(t)\mathbf{A}_s \cdot \hat{\mathbf{p}} + \frac{1}{2}f(t)\hat{\mathbf{p}} \cdot \mathbf{A}_s + \frac{1}{2}f(t)^2\|\mathbf{A}_s(\hat{\rho}, \hat{z})\|^2. \tag{5}$$

Moreover, `Fiend` supports all interaction operators of the form $\hat{W} = W(\hat{\rho}, \hat{z}, \hat{p}_\rho, \hat{p}_z, t)$.

The above interaction operators conserve the magnetic quantum number m, and by default, we simulate only the $m = 0$ subspace. Correspondingly, the coordinate space domain is a two-dimensional slice of the cylindrical coordinates, $\Omega_\infty = \{\rho \geq 0, z \in \mathbb{R}\}$, which we truncate for numerical simulations to a *finite* domain Ω [see Fig. 1(a)].

2.2. Time independent Schrödinger equation

On the domain Ω_{TI} the time-independent Schrödinger equation (TISE) can be written as

$$-\frac{1}{2\rho}\frac{\partial}{\partial\rho}\left(\rho\frac{\partial\psi_k}{\partial\rho}\right) - \frac{1}{2}\frac{\partial^2\psi_k}{\partial z^2} + V(\rho, z)\psi_k = E_k\psi_k, \tag{6}$$

where E_k, ψ_k is the kth eigenpair. This equation is accompanied by continuity boundary condition at $\rho = 0$,

$$\lim_{\rho \to 0+}\left(\rho\frac{\partial\psi}{\partial\rho}\right) = 0. \tag{7}$$

Elsewhere on the boundary $\partial\Omega_{\text{TI}}$ we can choose freely between zero Neumann boundary conditions (ZNBCs) and zero Dirichlet boundary conditions (ZDBCs) – depending on the requirements of our model. For example, in Fig. 1(a) where we have imposed ZNBC on the upper boundary of the simulation domain, Γ_N, and ZDBC on the arc Γ_D.

To discretize TISE, we begin by looking for eigenpairs $E_k \in \mathbb{R}, \psi_k \in F$ of the weak form corresponding to Eq. (6) and the boundary conditions described above. The weak form is given by

$$-\frac{1}{2}\int_{\Omega_{\text{TI}}} \sum_{\alpha=\rho,z} \frac{\partial\chi}{\partial\alpha}\frac{\partial\psi_k}{\partial\alpha}\rho\,\mathrm{d}\rho\,\mathrm{d}z$$
$$+ \int_{\Omega_{\text{TI}}} \chi V(\rho, z)\psi_k\rho\,\mathrm{d}\rho\,\mathrm{d}z \tag{8}$$
$$= E_k \int_{\Omega_{\text{TI}}} \chi\psi_k\rho\,\mathrm{d}\rho\,\mathrm{d}z \,\, \forall \chi \in F,$$

where $F = \{\psi \in H^1(\Omega_{\text{TI}}) \,|\, \psi(\Gamma_D) = 0\}$ is the standard Sobolev space on Ω_{TI} of real-valued L^2-integrable functions with L^2 integrable first derivatives. Notice also that F includes ZDBCs, and the natural inner product is

$$\langle\chi|\psi\rangle = \int_{\Omega_{\text{TI}}} \chi(\rho, z)^*\psi(\rho, z)\,\rho\,\mathrm{d}\rho\,\mathrm{d}z. \tag{9}$$

Furthermore, we must restrict ourselves to a finite dimensional approximation of F. First, the simulation domain Ω_{TI} is described using an unstructured triangular mesh which supports arbitrary refinement [see, e.g., Fig. 1(b)]. Next, we construct a basis of continuous low-order Lagrange polynomials ϕ_i, each with compact support on the mesh elements. This basis spans a finite dimensional function space

$$F_h = \text{span}\{\phi_i\}_{i=0}^{N-1} \subset F \tag{10}$$

where the TISE weak form [Eq. (8)] can be written as a finite dimensional generalized Hermitian eigenvalue equation

$$(\mathbf{T} + \mathbf{V})\boldsymbol{\psi}_k = \mathbf{S}E_k\boldsymbol{\psi}_k. \tag{11}$$

Here $\boldsymbol{\psi}_k$ is a vector of the real-valued expansion coefficients of the kth eigenstate, E_k the correspond-

ing eigenvalue, and

$$S_{ij} = \langle \phi_i | \phi_j \rangle, \qquad (12)$$

$$T_{ij} = -\frac{1}{2} \sum_{\alpha=\rho,z} \left\langle \frac{\partial \phi_i}{\partial \alpha} \Big| \frac{\partial \phi_j}{\partial \alpha} \right\rangle, \text{ and} \qquad (13)$$

$$V_{ij} = \langle \phi | V(\rho, z) \phi_j \rangle \qquad (14)$$

are the overlap, kinetic energy, and potential energy matrices, respectively. Note that since the static potential of the system is implemented only via Eq. (14) which involves integration, we can use any static potential for which Eq. (14) is finite, e.g., the Coulomb potential.

2.3. Time dependent Schrödinger equation

In a similar manner as with TISE, TDSE can be discretized as

$$i\mathbf{S}\dot{\boldsymbol{\psi}}(t) = (\mathbf{T} + \mathbf{V} + \mathbf{W})\,\boldsymbol{\psi}(t), \qquad (15)$$

where $\boldsymbol{\psi}(t)$ is a vector of the *complex-valued* expansion coefficients $c_i(t)$ of the wave function $\psi(\rho, z, t) = \sum c_i(t)\phi_i(\rho, z)$. $\mathbf{S}$, $\mathbf{T}$, and $\mathbf{V}$ are constructed the same way as for TISE (except we add an imaginary absorbing potential to $\mathbf{V}$), and the light-matter interaction matrix is given by

$$W_{ij} = \left\langle \phi_i \Big| W_s \left(\rho, z, \frac{\partial}{\partial \rho}, \frac{\partial}{\partial z}, t \right) \Big| \phi_j \right\rangle \qquad (16)$$

The basis of the discrete function space for TDSE can, in general, be different from the one used for TISE. This is useful in practice since often the stationary states can be computed in a much smaller simulation domain than needed for an accurate description of the TDSE. We can change the function space by three methods: (1) using a larger simulation domain for the TDSE, (2) refining the mesh according to the requirements of the TDSE simulation, and (3) change the degree of our basis functions according to the required accuracy.

For evolving the discretized state, i.e., the expansion coefficients $\boldsymbol{\psi}(t)$ according to Eq. (15), Fiend implements two approximations of the time evolution operator: the Crank-Nicolson (CN) method [44] and the generalized-α method [45].

The Crank-Nicolson method can be written as

$$\psi(t + \Delta t) = \left[\mathbf{S} + \frac{\mathrm{i}\Delta t}{2} \mathbf{H} \left(t + \frac{\Delta t}{2} \right) \right]^{-1}$$
$$\left[\mathbf{S} - \frac{\mathrm{i}\Delta t}{2} \mathbf{H} \left(t + \frac{\Delta t}{2} \right) \right] \psi(t) + \mathcal{O}(\Delta t^2),$$
$$(17)$$

and it is a unitary time-reversible transformation [44] as long as the matrix inversion is performed to a sufficient accuracy.

The generalized-α method was originally developed for integrating the equations of motion arising from fluid dynamics [45], but we have successfully applied it to the TDSE integration. To the best of our knowledge, the α-method has not been proven to be unitary nor is it exactly time-reversible. Nevertheless, we have achieved accuracies comparable to the CN method – and in the case of extremely inhomogeneous vector potentials, the generalized-α method seems to be significantly more stable than CN.

2.4. Incorporation of boundary conditions

The continuity boundary condition at $\rho = 0$ and ZNBCs are automatically included by dropping the boundary integrals when deriving Eqs. (8) and (13). However, the ZDBCs must be included in the system matrices via the following modifications: the rows and columns of $\boldsymbol{S}$, $\boldsymbol{T}$, $\boldsymbol{V}$, and $\boldsymbol{W}$ corresponding to the degrees of freedom (DOFs) on the boundary Γ_D with ZDBCs set to zero. Only for $\boldsymbol{S}$ we must set the corresponding diagonal elements to one to ensure invertibility. Note that it's crucial to zero out not only the rows but also the corresponding columns of the matrices to retain hermiticity.

Embedding ZNBCs via the vanishing boundary terms when deriving Eqs. (8) and (13) causes a practical issue: Some operators such as $\hat{p}_z = -\mathrm{i}\partial_z$, $\hat{p}_\rho = -\mathrm{i}\left(\partial_\rho + \frac{1}{2\rho}\right)$, and $\boldsymbol{A} \cdot \hat{\boldsymbol{p}} + \hat{\boldsymbol{p}} \cdot \boldsymbol{A}$ become *non-Hermitian*. Minor adjustments at the boundaries can remedy this, namely,

$$\hat{p}_\rho \rightarrow -\mathrm{i} \left[\partial_\rho + \frac{1}{2\rho} - \frac{1}{2}\delta(\boldsymbol{r} - \Gamma_N)n_\rho(\boldsymbol{r}) \right], \qquad (18)$$

$$\hat{p}_z \rightarrow -\mathrm{i} \left[\partial_z - \frac{1}{2}\delta(\boldsymbol{r} - \Gamma_N)n_z(\boldsymbol{r}) \right], \text{ and} \qquad (19)$$

$$\frac{1}{2}(\boldsymbol{A} \cdot \hat{\boldsymbol{p}} + \hat{\boldsymbol{p}} \cdot \boldsymbol{A}) \rightarrow \frac{1}{2}(\boldsymbol{A} \cdot \hat{\boldsymbol{p}} + \hat{\boldsymbol{p}} \cdot \boldsymbol{A})$$
$$- \frac{\mathrm{i}}{2}\boldsymbol{A} \cdot \boldsymbol{n}(\boldsymbol{r})\delta(\boldsymbol{r} - \Gamma_N), \qquad (20)$$

where $\boldsymbol{n}(\boldsymbol{r}) = [n_\rho(\boldsymbol{r}), n_z(\boldsymbol{r})]^T$ is the outwards facing unit normal on the simulation domain boundary at point $\boldsymbol{r}$. This trick has been previously used to obtain the correct Hermitian operators in hyperspherical coordinates [43].

Another aspect to consider is the weak enforcement of ZNBCs compared to ZDBCs. If there was

significant electron density on the Neumann boundary Γ_N, it will start to oscillate and eventually violate the ZNBC. This can be remedied when operating with matrix inverses as in the second step of the CN-propagator. We can add an extra error term, e.g., of the form

$$\gamma \sum_i \left| \int_{\Gamma_N} \chi_i^*(\rho, z) \nabla \psi(\rho, z, t) \cdot \mathrm{d}\boldsymbol{S} \right|^2, \gamma \in \mathbb{R} \quad (21)$$

to the error estimator in our numerical implementation. This will remove – or at least lessen – the issue arising from the weak enforcement of ZNBCs.

3. Implementation

3.1. Overview

We have implemented finite element discretization of TISE and TDSE in the Python software package `Fiend` following the recipes of Sec. 2. `Fiend` is written in Python 3.6 and it utilizes reliable libraries commonly pre-installed in high-performance clusters and supercomputers. For sparse linear algebra we use the Portable, Extensible Toolkit for Scientific Computation (`PETSc`) [46–48] and the Scalable Library for Eigenvalue Problem Computations (`SLEPc`) [49, 50]. Meshing and other standard FEM-related parts are built on top of the components of the FEniCS project [34–40], and filesystem IO is largely based on `HDF5` [51] via `h5py` [52, 53]. In postprocessing and visualization we use also `numpy` [54], `scipy` [55], and `matplotlib` [56]. `Fiend` is parallelized using `MPI` via `mpi4py` [57–59].

The `PETSc`-dependency is slightly complicated as `PETSc` needs to be compiled either with real number support (x)or with complex number support. `Fiend`, on the other hand, needs real numbers for TISE and post-processing but complex numbers for propagation. Consequently, the user must install *both* the real and complex versions of `PETSc`/`SLEPc` stacks and switch between these for different simulation steps. This factitious requirement for two different installations of `PETSc` can be revisited in the future upon the completion of `DOLFIN-X` [60] and `FFC-X` [61] including the merge of complex number support in `FEniCS` [62],

To ease installation, we provide a Docker [42] container with `Fiend` and all its dependencies preinstalled, see https://hub.docker.com/r/solanpaa/fiend/ for more details. In the same spirit, `Fiend` is also available in the Python Package Index [41] and can be installed with `pip` [63].

Load `FEniCS`/`PETSc` with real numbers

1. Solve TISE

- Mesh Ω_{TI}
- Assemble matrices
- Solve eigenproblem
- Save results

2. Preparation step

- Mesh Ω_{TD}
- Remesh stationary states if necessary
- Assemble matrices
- Save matrices and states and convert them to complex numbers

Switch to `PETSc` with complex numbers

3. Propagation

- Load matrices and states from step 2
- Construct time-dependent Hamiltonian
- Compute $\boldsymbol{\psi}(t)$ and save data

Switch to `PETSc` with real numbers

4. Analysis

- Post-processing
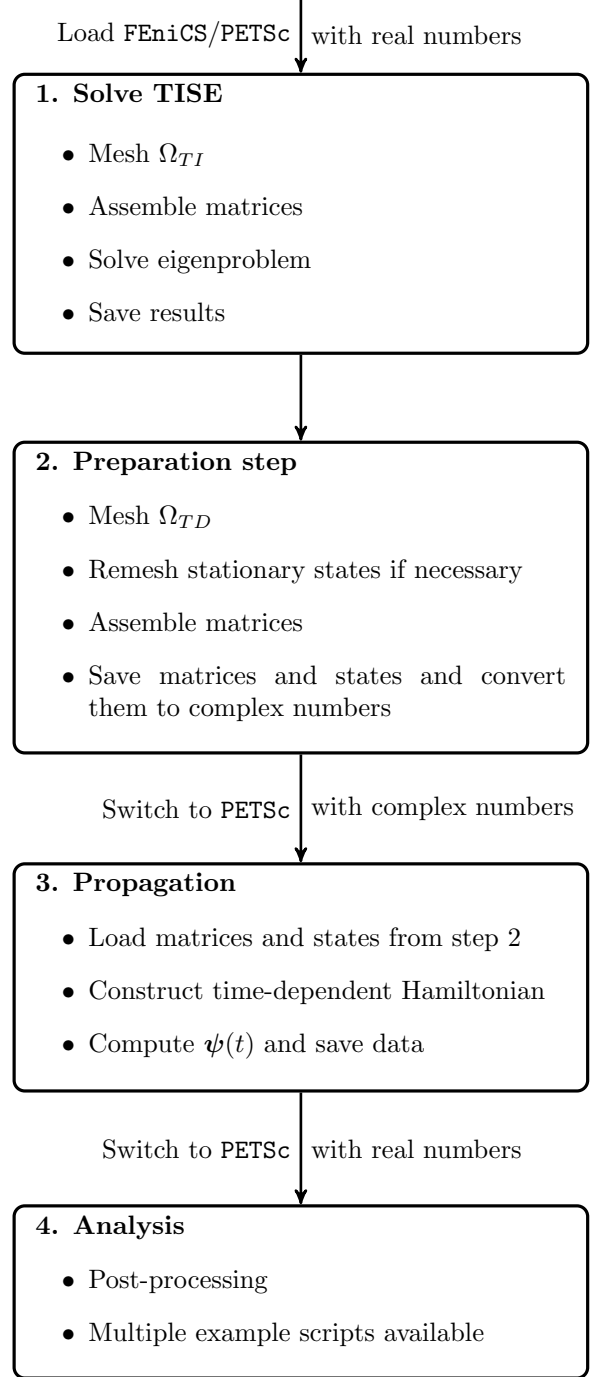- Multiple example scripts available

Figure 2: Typical simulation structure with `Fiend` together with the required changes to the runtime environment between the steps.

3.2. Library usage and numerical methods

A typical usage of the `Fiend` suite is demonstrated in Fig. 2. First we solve the TISE to obtain a set of stationary states. This includes meshing the domain for the time independent simulation and assembling the system matrices (12)–(14) with appropriate boundary conditions (see Sec. 2.4).

The TISE eigenproblem, Eq. (11), is solved (by default) with Rayleigh Quotient Conjugate Gradient (RQCG) method combined with classical Gram-Schmidt orthogonalization of the Krylov subspace basis with adaptive iterative refinement for increased numerical stability. RQCG is a variational method which essentially minimizes the Rayleigh quotient

$$\frac{\boldsymbol{\psi}_k^\dagger \boldsymbol{H}_0 \boldsymbol{\psi}_k}{\boldsymbol{\psi}_k^\dagger \boldsymbol{S} \boldsymbol{\psi}_k} \tag{22}$$

of a desired number of orthonormal vectors with respect to the bilinear product induced by $\boldsymbol{S}$ [50]. Upon convergence, this corresponds to k smallest real eigenpairs of TISE.

Step 2 (see Fig. 2) is to prepare the discrete description of the TDSE. This includes meshing the domain for the time dependent problem, interpolating stationary states to the new mesh, and also assembly of the system matrices (12)–(14) and the time-independent part(s) of Eq. (16).

The propagation in Step. 3 (Fig. 2) requires the user to load the environment with complex number `PETSc`. The time-dependent Hamiltonian is constructed from the matrices computed in the previous step, and the discrete TDSE (15) is solved with `PETSc`'s propagators. We also provide code templates for easy implementation of new propagators.

Finally, we provide a set of example scripts for postprocessing. These scripts include, e.g., visualization of the integrated electron density in time and temporal shape of the laser field, but there are also scripts to compute more complex observables such as the high-harmonic and photoelectron spectra.

All the numerical methods in `Fiend` depend on efficient sparse linear algebra operations implemented in `PETSc`. By default, matrix inversions and the solution of linear equations are carried out with Generalized Minimal RESidual method (GMRES) [64], where the convergence criterion is computed with respect to the norm induced by the inner product in Eq. (9). Unfortunately, the `PETSc` linear algebra backend does not allow us to build the basis of the Krylov subspace with respect to a custom inner product. However, we have found the basis built with the standard Hermitian inner product to be sufficient performance-wise. Direct solvers such as `SuperLU_DIST` [65–67] and `MUMPS` [68, 69] are supported only with matrices that can be explicitly constructed without too high cost.

3.3. Note on meshing

We provide a way to generate meshes with arbitrary refinement in the coordinate space. The user should supply a function returning the maximum allowed cell circumradius[3] at coordinate $\boldsymbol{r}$, and the mesh can be refined until all cells of the mesh have circumradius below the desired one.

By default, the maximum cell circumradius is given by [70]

$$\max \mathrm{CR}(\boldsymbol{r}) = \mathrm{CR}_{\mathrm{asymp}} \left[1 - \left(1 - \frac{\mathrm{CR}_{\mathrm{ref}}}{\mathrm{CR}_{\mathrm{asymp}}} \right) \right.$$
$$\left. \times \frac{R_{\mathrm{ref}}}{r} \tanh \left(\frac{r}{R_{\mathrm{ref}}} \right) \exp \left(-\frac{r^2}{R_{\mathrm{trans}}^2} \right) \right]. \tag{23}$$

For $r < R_{\mathrm{ref}}$ the cell circumradii are below $\mathrm{CR}_{\mathrm{ref}}$, and as $r \to R_{\mathrm{trans}}$, the maximum cell circumradius increases monotonically to $\mathrm{CR}_{\mathrm{asymp}}$.

4. Examples and test cases

4.1. Field-free propagation

To assess the stability and accuracy of the numerical methods implemented in `Fiend`, we compute a field-free propagation of an electron prepared in a superposition of Hydrogen 1s and 2s states,

$$\psi(t = 0) = \frac{1}{\sqrt{2}} \left(\psi_{1s} + \psi_{2s} \right). \tag{24}$$

This state is propagated up to 100 a.u. of time. The simulation domain up to $R = 30$ a.u. is meshed using the default refinement introduced in Sec. 3.3 with $R_{\mathrm{ref}} = 4$ a.u., $R_{\mathrm{ref}} = 10$ a.u., $\mathrm{CR}_{\mathrm{ref}} = 0.01$ a.u. and $\mathrm{CR}_{\mathrm{asymp}} = 0.5$ a.u. We assess the accuracy of the numerical solution by comparing the simulated state to the exact result:

$$\left| 1 - \left| \langle \psi_{\mathrm{exact}} | \psi_{\mathtt{Fiend}} \rangle \right|^2 \right|. \tag{25}$$

---

[3]Radius of the smallest circle enclosing the given triangular cell.

The stability is investigated by evaluating how much the norm of the state differs from unity, i.e.,

$$\left|1 - \|\psi_{\mathtt{Fiend}}\|^2\right|. \tag{26}$$

Both the α-propagator and the CN propagator reach an accuracy of 0.00524 % for the overlap with the time-step 0.05 a.u., and the norm of the simulated state deviates from unity only by $3.22 \cdot 10^{-5}$ %. This demonstrates the applicability of the propagators for the simulation of FE-discretized TDSE in cylindrical coordinates.

4.2. High-harmonic generation

Next we demonstrate the applicability of Fiend to simulate high-order harmonic generation (HHG). According to the three-step model [71, 72], an atomic electron is excited to continuum and driven to oscillate around the atom [73]. Upon recombination with the parent ion, the excess energy is released as high-energy photons.

Full 3D simulation of HHG via TDSE is a difficult task due to the long extent of the electron wave function when driven by a short laser pulse. In order to obtain a decent description of the single-atom response for HHG, a large simulation domain is needed. We setup a simulation domain of radius 800 a.u. with a more refined mesh at the origin and gradually sparser mesh for larger radii.

The intensity spectrum of the single-atom response is proportional to the square of the Fourier transform of the dipole acceleration [74],

$$S(\omega) \propto \left|\left\langle \ddot{D}_z \right\rangle (\omega)\right|^2, \tag{27}$$

where we compute the dipole acceleration via Ehrenfest's theorem [75], i.e.,

$$\left\langle \ddot{D}_z \right\rangle (t) = \left\langle \psi(t) \left| -\frac{\partial V}{\partial z} \right| \psi(t) \right\rangle. \tag{28}$$

In Fig. 3 we show the computed high-order harmonic spectrum computed for a hydrogen atom under a laser pulse with carrier frequency corresponding to 800 nm, intensity full width at half maximum 4.8 fs, and electric field peak intensity 36 GV/m. The spectrum is a typical HHG spectrum from a femtosecond laser pulse, and it ends at the cutoff energy ~ 2 a.u. in a decent agreement with the three-step model's cutoff ~ 1.7 a.u. [71, 72].
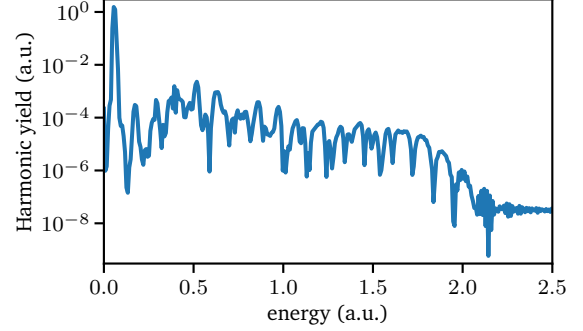

Figure 3: High-order harmonic spectrum of a hydrogen atom under a few-cycle femtosecond laser pulse.

4.3. Metal nanotips and inhomogeneous fields

Metal nanotips have recently attracted attention due to their ability to enhance the laser electric field at the tip apex via plasmonic effects (see, e.g., [19–28]). Fiend is capable of computing single-electron dynamics of these metal nanotapers including correctly the interaction with the *inhomogeneous* plasmon enhanced laser field.

The first step is to compute the plasmon enhanced near field. Here we consider a gold nanotip with apex radius 7 nm and full opening angle of $20°$ as demonstrated in Fig. 4(a). For typical laser wavelengths, such as 800 nm used here, a quasistatic description of the laser vector potential is applicable, i.e.,

$$\mathbf{A}(\mathbf{r}, t) = \mathbf{A}_s(\mathbf{r}) f(t), \tag{29}$$

where the spatial form $\mathbf{A}_s$ is the same as for the electric field

$$\mathbf{A}_s(\mathbf{r}) = \mathbf{E}(\mathbf{r}) = -\nabla U(\mathbf{r}). \tag{30}$$

The electrostatic potential $U(\mathbf{r})$ can be computed from the Poisson equation

$$-\nabla \cdot [\epsilon(\mathbf{r}) \nabla U(\mathbf{r})] = 0, \tag{31}$$

where $\epsilon(\mathbf{r})$ is the dielectric function of the material at position $\mathbf{r}$. Inside the gold nanotip the complex dielectric function (at 800 nm) is $\epsilon_{\mathrm{Au}} = -24.061 + 1.5068\,\mathrm{i}$ [76, 77] and at vacuum $\epsilon_{\mathrm{vac}} = 1$.

We can describe Eq. (33) using cylindrical coordinates. The interface condition

$$\epsilon_{\mathrm{Au}} \nabla u|_{\boldsymbol{r} \to \Gamma_{\mathrm{tip\text{-}vac}}} - \epsilon_{\mathrm{vac}} \nabla u|_{\boldsymbol{r} \to \Gamma_{\mathrm{vac\text{-}tip}}} = 0 \tag{32}$$
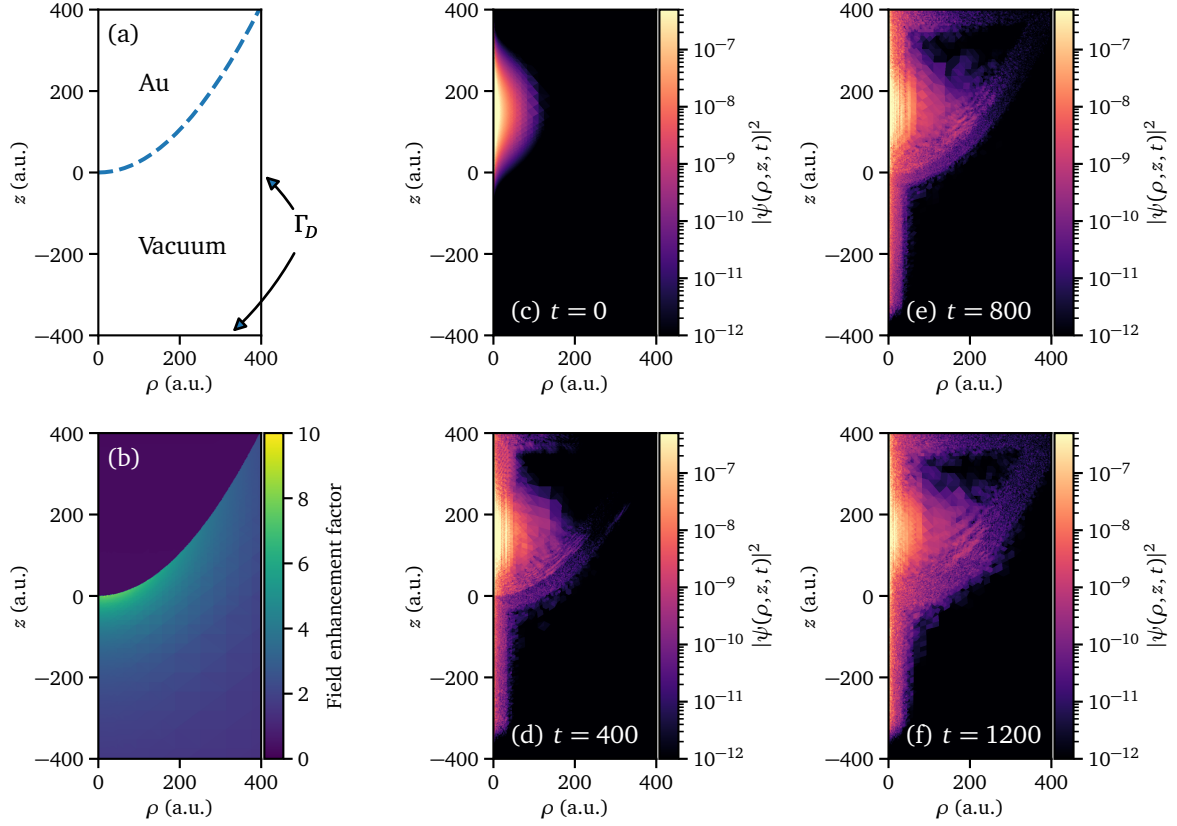
Figure 4: (a) Setup of the nanotip geometry where we have a sharp boundary between the gold nanotip and vacuum. (b) Computed spatial enhancement profile of the electric near field in a quasistatic approximation. (c)-(f) Snapshots of the electron density during time evolution.

gives rise to the weak form

$$
\epsilon_{\text{vac}} \underbrace{\int \nabla v^*(\mathbf{r}) \cdot \nabla u(\mathbf{r}) \, \rho \, d\rho \, dz}_{\text{vacuum}}
$$
$$
+ \epsilon_{\text{Au}} \underbrace{\int \nabla v^*(\mathbf{r}) \cdot \nabla u(\mathbf{r}) \, \rho \, d\rho \, dz}_{\text{tip}} = 0 \; \forall v \in \hat{U}
\tag{33}
$$

with the test function space

$$
\hat{U} = \{ v : \Omega \to \mathbb{C} \mid \text{Re}(v), \, \text{Im}(v) \in H^1(\Omega) : \\
v(\Gamma_D) = 0 \}
\tag{34}
$$

and the trial function space

$$
U = \{ u : \Omega \to \mathbb{C} \mid \text{Re}(u), \, \text{Im}(u) \in H^1(\Omega) : \\
u(\mathbf{r} \in \Gamma_D) = z \}.
\tag{35}
$$

The simulation domain is a rectangular area, and the ZDBCs described above are imposed on the external boundaries denoted by Γ_D. Note that Fig. 4(a) demonstrates the domain and boundaries for the quantum simulation – for Poisson equation we employ a much larger simulation domain. The equation (33) can be solved easily with the FEniCS FEM-suite and we provide an example script at demos/nanotip/1_near_field.py

The computed spatial distribution of the field, $\|A_s(x, y, z)\|$, is demonstrated in Fig. 4(b). The field is at its maximum at the nanotip apex and decays rapidly with the distance to the tip. We note that this quasistatic solution provides field-enhancement factor of $f \approx 10$ which is comparable with experimental results [78].

In TDSE, we use a potential well for the static potential,

$$
U(\mathbf{r}) = \begin{cases} 0, & \text{vacuum} \\ -\Phi, & \text{inside the tip}, \end{cases}
\tag{36}
$$

where $\Phi = 5.31 \, \text{eV}$ is the work function of Au (111)-

surface [79].

We prepare the electron as a Gaussian wave packet [Fig. 4(c)] and assemble the system matrices – *including the inhomogeneous field* – with `demos/nanotip/3_prepare_tdse.py`. The preparation step is slightly more complex than for linearly polarized vector potentials, but the provided example script should work as a template for further expansions.

We use a 800 nm pulse with 15 fs full width at half maximum for the Gaussian envelope. The maximum electric field amplitude without nanostructure enhancement is set to 30 GV/nm. Snapshots of the electron density during propagation are shown in Figs. 4(c)-(f). As expected, the electron emission is concentrated at the tip apex where the field enhancement is at its highest. Only a small percentage ($\sim 2\%$) of the electron density is absorbed at the simulation box boundary.

5. Summary

We have presented `Fiend` – a versatile solver for single-particle quantum dynamics in cylindrically symmetric systems. This Python package provides an easy path for the study of nonlinear strong-field phenomena in atoms and nanostructures under both homogeneous and *inhomogeneous* vector potentials. Moreover, the finite element discretization used by `Fiend` can adapt to complicated system geometries. The package is parallelized using `MPI`, and much of the high-performance computing is delegated to state-of-the-art numerical libraries. `Fiend` is modular and easy to extend, and we provide comprehensive documentation for the program code.

We have demonstrated the capabilities of `Fiend` by simulating two different scenarios. The simulation of high-order harmonic generation from a laser-driven hydrogen atom shows that `Fiend` can be applied to traditional strong-field phenomena in atoms. Furthermore, a simulation of the photoionization of a gold nanotip demonstrates `Fiend`'s suitability for studying plasmon-enriched strong-field phenomena in nanostructures.

Acknowledgements

References

[1] N. Suárez, A. Chacón, J. A. Pérez-Hernández, J. Biegert, M. Lewenstein, M. F. Ciappina, High-order-harmonic generation in atomic and molecular systems, Phys. Rev. A 95 (2017) 033415. `doi:10.1103/PhysRevA.95.033415`.

[2] N. I. Shvetsov-Shilovski, M. Lein, L. B. Madsen, E. Räsänen, C. Lemell, J. Burgdörfer, D. G. Arbó, K. Tőkési, Semiclassical two-step model for strong-field ionization, Phys. Rev. A 94 (2016) 013415. `doi:10.1103/PhysRevA.94.013415`.

[3] A. Galstyan, O. Chuluunbaatar, A. Hamido, Y. V. Popov, F. Mota-Furtado, P. F. O'Mahony, N. Janssens, F. Catoire, B. Piraux, Reformulation of the strong-field approximation for light-matter interactions, Phys. Rev. A 93 (2016) 023422. `doi:10.1103/PhysRevA.93.023422`.

[4] M. Klaiber, J. c. v. Daněk, E. Yakaboylu, K. Z. Hatsagortsyan, C. H. Keitel, Strong-field ionization via a high-order coulomb-corrected strong-field approximation, Phys. Rev. A 95 (2017) 023403. `doi:10.1103/PhysRevA.95.023403`.

[5] A.-T. Le, H. Wei, C. Jin, C. D. Lin, Strong-field approximation and its extension for high-order harmonic generation with mid-infrared lasers, Journal of Physics B: Atomic, Molecular and Optical Physics 49 (5) (2016) 053001. `doi:10.1088/0953-4075/49/5/053001`.

[6] S. Majorosi, M. G. Benedict, A. Czirják, Improved one-dimensional model potentials for strong-field simulations, Phys. Rev. A 98 (2018) 023401. `doi:10.1103/PhysRevA.98.023401`.

[7] L. Wang, G.-L. Wang, Z.-H. Jiao, S.-F. Zhao, X.-X. Zhou, High-order harmonic generation of li + with combined infrared and extreme ultraviolet fields, Chinese Physics B 27 (7) (2018) 073205. `doi:10.1088/1674-1056/27/7/073205`.

[8] Z. Abdelrahman, M. A. Khokhlova, D. J. Walke, T. Witting, A. Zair, V. V. Strelkov, J. P. Marangos, J. W. G. Tisch, Chirp-control of resonant high-order harmonic generation in indium ablation plumes driven by intense few-cycle laser pulses, Opt. Express 26 (12) (2018) 15745–15758. `doi:10.1364/OE.26.015745`.

[9] A.-T. Le, H. Wei, C. Jin, V. N. Tuoc, T. Morishita, C. D. Lin, Universality of returning electron wave packet in high-order harmonic generation with midinfrared laser pulses, Phys. Rev. Lett. 113 (2014) 033001. `doi:10.1103/PhysRevLett.113.033001`.

[10] C. Hofmann, A. S. Landsman, A. Zielinski, C. Cirelli, T. Zimmermann, A. Scrinzi, U. Keller, Interpreting electron-momentum distributions and nonadiabaticity in strong-field ionization, Phys. Rev. A 90 (2014) 043406. `doi:10.1103/PhysRevA.90.043406`.

[11] V. P. Majety, A. Zielinski, A. Scrinzi, Mixed gauge in strong laser-matter interaction, Journal of Physics B: Atomic, Molecular and Optical Physics 48 (2) (2015) 025601. `doi:10.1088/0953-4075/48/2/025601`.

[12] A. M. Sayler, M. Arbeiter, S. Fasold, D. Adolph, M. Möller, D. Hoff, T. Rathje, B. Fetić, D. B. Milošević, T. Fennel, G. G. Paulus, Accurate determination of absolute carrier-envelope phase dependence using photo-

ionization, Opt. Lett. 40 (13) (2015) 3137–3140. `doi: 10.1364/OL.40.003137`.

[13] X. Y. Lai, W. Quan, S. G. Yu, Y. Y. Huang, X. J. Liu, Suppression in high-order above-threshold ionization: destructive interference from quantum orbits, Journal of Physics B: Atomic, Molecular and Optical Physics 51 (10) (2018) 104003. `doi:10.1088/1361-6455/aabc83`.

[14] D. Zille, D. Adolph, M. Möller, A. M. Sayler, G. G. Paulus, Chirp and carrier-envelope-phase effects in the multiphoton regime: measurements and analytical modeling of strong-field ionization of sodium, New Journal of Physics 20 (6) (2018) 063018. `doi:10.1088/1367-2630/aac3ca`.

[15] J. A. Schuller, E. S. Barnard, W. Cai, Y. C. Jun, J. S. White, M. L. Brongersma, Plasmonics for extreme light concentration and manipulation, Nature Materials 9 (2010) 193 EP. `doi:10.1038/nmat2630`.

[16] L. Ortmann, J. A. Pérez-Hernández, M. F. Ciappina, J. Schötz, A. Chacón, G. Zeraouli, M. F. Kling, L. Roso, M. Lewenstein, A. S. Landsman, Emergence of a higher energy structure in strong field ionization with inhomogeneous electric fields, Phys. Rev. Lett. 119 (2017) 053204. `doi:10.1103/PhysRevLett.119.053204`.

[17] L. Ortmann, A. S. Landsman, Analysis of the higher-energy structure in strong-field ionization with inhomogeneous electric fields, Phys. Rev. A 97 (2018) 023420. `doi:10.1103/PhysRevA.97.023420`.

[18] M. F. Ciappina, J. A. Pérez-Hernández, L. Ortmann, J. Schötz, A. Chacón, G. Zeraouli, M. F. Kling, L. Roso, M. Lewenstein, A. S. Landsman, Emergence of a higher energy structure in strong field ionization with inhomogeneous laser fields, in: High-Brightness Sources and Light-driven Interactions, Optical Society of America, 2018, p. HM4A.5. `doi:10.1364/HILAS.2018.HM4A.5`.

[19] B. Piglosiewicz, S. Schmidt, D. J. Park, J. Vogelsang, P. Groß, C. Manzoni, P. Farinello, G. Cerullo, C. Lienau, Carrier-envelope phase effects on the strong-field photoemission of electrons from metallic nanostructures, Nature Photonics 8 (2013) 37. `doi:10.1038/nphoton.2013.288`.

[20] M. Kruger, M. Schenk, M. Forster, P. Hommelhoff, Attosecond physics in photoemission from a metal nanotip, Journal of Physics B: Atomic, Molecular and Optical Physics 45 (7) (2012) 074006. `doi:10.1088/0953-4075/45/7/074006`.

[21] L. Wimmer, G. Herink, D. R. Solli, S. V. Yalunin, K. E. Echternkamp, C. Ropers, Terahertz control of nanotip photoemission, Nature Physics 10 (2014) 43. `doi:0.1038/nphys2974`.

[22] R. Bormann, S. Strauch, S. Schäfer, C. Ropers, An ultrafast electron microscope gun driven by two-photon photoemission from a nanotip cathode, Journal of Applied Physics 118 (17) (2015) 173105. `doi:10.1063/1.4934681`.

[23] G. M. Caruso, F. Houdellier, P. Abeilhou, A. Arbouet, Development of an ultrafast electron source based on a cold-field emission gun for ultrafast coherent tem, Applied Physics Letters 111 (2) (2017) 023101. `doi:10.1063/1.4991681`.

[24] B. Barwick, C. Corder, J. Strohaber, N. Chandler-Smith, C. Uiterwaal, H. Batelaan, Laser-induced ultrafast electron emission from a field emission tip, New Journal of Physics 9 (5) (2007) 142. `doi:10.1088/1367-2630/9/5/142`.

[25] G. Herink, D. R. Solli, M. Gulde, C. Ropers, Field-driven photoemission from nanostructures quenches the quiver motion, Nature 483 (2012) 190. `doi:10.1038/nature10878`.

[26] R. Bormann, M. Gulde, A. Weismann, S. V. Yalunin, C. Ropers, Tip-enhanced strong-field photoemission, Phys. Rev. Lett. 105 (2010) 147601. `doi:10.1103/PhysRevLett.105.147601`.

[27] B. Förg, J. Schötz, F. Süßmann, M. Förster, M. Krüger, B. Ahn, W. A. Okell, K. Wintersperger, S. Zherebtsov, A. Guggenmos, V. Pervak, A. Kessel, S. A. Trushin, A. M. Azzeer, M. I. Stockman, D. Kim, F. Krausz, P. Hommelhoff, M. F. Kling, Attosecond nanoscale near-field sampling, Nature Communications 7 (2016) 11717 EP –, article.
URL `https://doi.org/10.1038/ncomms11717`

[28] S. V. Yalunin, G. Herink, D. R. Solli, M. Krüger, P. Hommelhoff, M. Diehn, A. Munk, C. Ropers, Field localization and rescattering in tip-enhanced photoemission, Annalen der Physik 525 (1-2) L12–L18. `doi:10.1002/andp.201200224`.

[29] R. G. Hobbs, Y. Yang, P. D. Keathley, M. E. Swanwick, L. F. Velásquez-García, F. X. Kärtner, W. S. Graves, K. K. Berggren, High-density au nanorod optical field-emitter arrays, Nanotechnology 25 (46) (2014) 465304. `doi:10.1088/0957-4484/25/46/465304`.

[30] F. Kusa, K. E. Echternkamp, G. Herink, C. Ropers, S. Ashihara, Optical field emission from resonant gold nanorods driven by femtosecond mid-infrared pulses, AIP Advances 5 (7) (2015) 077138. `doi:10.1063/1.4927151`.

[31] A. Grubisic, E. Ringe, C. M. Cobley, Y. Xia, L. D. Marks, R. P. Van Duyne, D. J. Nesbitt, Plasmonic near-electric field enhancement effects in ultrafast photoelectron emission: Correlated spatial and laser polarization microscopy studies of individual ag nanocubes, Nano Letters 12 (9) (2012) 4823–4829. `doi:10.1021/nl302271u`.

[32] R. G. Hobbs, Y. Yang, A. Fallahi, P. D. Keathley, E. De Leo, F. X. Kärtner, W. S. Graves, K. K. Berggren, High-yield, ultrafast, surface plasmon-enhanced, au nanorod optical field electron emitter arrays, ACS Nano 8 (11) (2014) 11474–11482. `doi:10.1021/nn504594g`.

[33] A. Grubisic, V. Schweikhard, T. A. Baker, D. J. Nesbitt, Coherent multiphoton photoelectron emission from single au nanorods: The critical role of plasmonic electric near-field enhancement, ACS Nano 7 (1) (2013) 87–99. `doi:10.1021/nn305194n`.

[34] The FEniCS Project Version 1.5, Archive of Numerical Software 3. `doi:10.11588/ans.2015.100.20553`.

[35] A. Logg, K.-A. Mardal, G. Wells (Eds.), Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book, Vol. 84 of Lecture Notes in Computational Science and Engineering, Springer-Verlag Berlin Heidelberg, 2012. `doi:10.1007/978-3-642-23099-8`.

[36] A. Logg, G. N. Wells, `DOLFIN`: Automated finite element computing, ACM Transactions on Mathematical Software 37 (2010) 20. `doi:10.1145/1731022.1731030`.

[37] R. C. Kirby, A. Logg, A compiler for variational forms, ACM Transactions on Mathematical Software 32 (2006) 417–444. `doi:scho10.1145/1163641.1163644`.

[38] K. B. Ølgaard, G. N. Wells, `DOLFIN`: Automated finite element computing, ACM Transactions on Mathematical Software 37 (2010) 8. `doi:10.1145/1644001`.

1644009.

[39] Unified form language: A domain-specific language for weak formulations of partial differential equations, ACM Transactions on Mathematical Software 40 (2014) 9. `doi:10.1145/2566630`.

[40] R. C. Kirby, Algorithm 839: `FIAT`, a new paradigm for computing finite element basis functions, ACM Transactions on Mathematical Software 30 (2004) 502. `doi:10.1145/1039813.1039820`.

[41] J. Solanpää, `Fiend` in PyPI, [Online; accessed Nov. 19th 2018] (2018).
URL `https://pypi.org/project/fiend`

[42] Docker Inc., Enterprise container platform, [Online; accessed Nov. 19th 2018] (2018).
URL `https://www.docker.com`

[43] G. Paz, On the connection between the radial momentum operator and the hamiltonian in n dimensions, European Journal of Physics 22 (4) (2001) 337. `doi:10.1088/0143-0807/22/4/308`.

[44] J. Crank, P. Nicolson, A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type, Advances in Computational Mathematics 6 (1) (1996) 207–226. `doi:10.1007/BF02127704`.

[45] K. E. Jansen, C. H. Whiting, G. M. Hulbert, A generalized-α method for integrating the filtered navier–stokes equations with a stabilized finite element method, Computer Methods in Applied Mechanics and Engineering 190 (3) (2000) 305 – 319. `doi:0.1016/S0045-7825(00)00203-6`.

[46] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, H. Zhang, PETSc users manual, Tech. Rep. ANL-95/11 - Revision 3.9, Argonne National Laboratory (2018).

[47] S. Balay, W. D. Gropp, L. C. McInnes, B. F. Smith, Efficient management of parallelism in object oriented numerical software libraries, in: E. Arge, A. M. Bruaset, H. P. Langtangen (Eds.), Modern Software Tools in Scientific Computing, Birkhäuser Press, 1997, pp. 163–202.

[48] L. D. Dalcin, R. R. Paz, P. A. Kler, A. Cosimo, Parallel distributed computing using `Python`, Advances in Water Resources 34 (9) (2011) 1124 – 1139, new Computational Methods and Software Tools. `doi:10.1016/j.advwatres.2011.04.013`.

[49] V. Hernandez, J. E. Roman, V. Vidal, `SLEPc`: A scalable and flexible toolkit for the solution of eigenvalue problems, ACM Trans. Math. Software 31 (3) (2005) 351–362. `doi:10.1145/1089014.1089019`.

[50] J. E. Roman, C. Campos, E. Romero, A. Tomas, `SLEPc` users manual, Tech. Rep. DSIC-II/24/02 - Revision 3.9, D. Sistemes Informàtics i Computació, Universitat Politècnica de València (2018).

[51] The HDF Group, Hierarchical Data Format, version 5 (2018).
URL `http://www.hdfgroup.org/HDF5/`

[52] HDF5 for Python, [Online; accessed Nov. 19th 2018].
URL `http://www.h5py.org`

[53] A. Collette, Python and HDF5, O'Reilly, 2013.

[54] T. E. Oliphant, A guide to `NumPy`, 2nd Edition, CreateSpace Independent Publishing Platform, 2015.

[55] E. Jones, T. Oliphant, P. Peterson, et al., `SciPy`: Open source scientific tools for Python, [Online; accessed Nov. 19th 2018] (2001–).
URL `http://www.scipy.org/`

[56] J. D. Hunter, `Matplotlib`: A 2d graphics environment, Computing in Science & Engineering 9 (3) (2007) 90–95. `doi:10.1109/MCSE.2007.55`.

[57] L. Dalcin, P. Kler, R. Paz, A. Cosimo, Parallel distributed computing using `Python`, Advances in Water Resources 34 (2011) 1124–1139. `doi:10.1016/j.advwatres.2011.04.013`.

[58] MPI for `Python`: performance improvements and `MPI-2` extensions, Journal of Parallel and Distributed Computing 68 (2008) 655–662. `doi:10.1016/j.jpdc.2007.09.005`.

[59] MPI for `Python`, Journal of Parallel and Distributed Computing 65 (2005) 1108–1115. `doi:10.1016/j.jpdc.2005.03.010`.

[60] Fenics/`dolfinx`: Next generation FEniCS problem solving environment, [Online; accessed Nov. 19th 2018] (2018–).
URL `https://github.com/FEniCS/dolfinx`

[61] FeniCS/`ffcx`: Next generation FEniCS form compuler, [Online; accessed Nov. 19th 2018] (2018–).
URL `https://github.com/FEniCS/ffcx`

[62] I. Baratta, Complex number support in FEniCS, [Online; accessed Nov. 19th 2018] (2018).
URL `https://gist.github.com/IgorBaratta/c7ca5252834f2c70efe0d233a3acecb4`

[63] Python Software Foundation, Installing `Python` modules, [Online; accessed Nov. 19th 2018] (2018).
URL `https://docs.python.org/3/installing/`

[64] Y. Saad, M. H. Schultz, Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (1983) 856–869. `doi:10.1137/0907058`.

[65] X. S. Li, An overview of `SuperLU`: Algorithms, implementation, and user interface, ACM Trans. Math. Softw. 31 (3) (2005) 302–325. `doi:10.1145/1089014.1089017`.

[66] X. Li, J. Demmel, J. Gilbert, iL. Grigori, M. Shao, I. Yamazaki, `SuperLU` Users' Guide, Tech. Rep. LBNL-44289, Lawrence Berkeley National Laboratory (September 1999).
URL `http://crd.lbl.gov/~xiaoye/SuperLU/`

[67] X. S. Li, J. W. Demmel, `SuperLU_DIST`: A scalable distributed-memory sparse direct solver for unsymmetric linear systems, ACM Trans. Mathematical Software 29 (2) (2003) 110–140. `doi:10.1145/779359.779361`.

[68] P. R. Amestoy, I. S. Duff, J. Koster, J.-Y. L'Excellent, A fully asynchronous multifrontal solver using distributed dynamic scheduling, SIAM Journal on Matrix Analysis and Applications 23 (1) (2001) 15–41. `doi:10.1137/S0895479899358194`.

[69] P. R. Amestoy, A. Guermouche, J.-Y. L'Excellent, S. Pralet, Hybrid scheduling for the parallel solution of linear systems, Parallel Computing 32 (2) (2006) 136–156. `doi:10.1016/j.parco.2005.07.004`.

[70] F. m. c. Gygi, G. Galli, Real-space adaptive-coordinate electronic-structure calculations, Phys. Rev. B 52 (1995) R2229–R2232. `doi:10.1103/PhysRevB.52.R2229`.
URL `https://link.aps.org/doi/10.1103/PhysRevB.52.R2229`

[71] J. L. Krause, K. J. Schafer, K. C. Kulander, High-order harmonic generation from atoms and ions in the high

intensity regime, Phys. Rev. Lett. 68 (1992) 3535–3538. `doi:10.1103/PhysRevLett.68.3535`.

[72] M. Lewenstein, P. Balcou, M. Y. Ivanov, A. L'Huillier, P. B. Corkum, Theory of high-harmonic generation by low-frequency laser fields, Phys. Rev. A 49 (1994) 2117–2132. `doi:10.1103/PhysRevA.49.2117`.

[73] M. Lewenstein, A. L'Huillier, Principles of single atom physics: High-order harmonic generation, above-threshold ionization and non-sequential ionization, in: T. Brabec (Ed.), Strong Field Laser Physics, Springer, New York, USA, 2008, pp. 147–184. `doi:10.1007/978-0-387-34755-4`.

[74] J. A. Pérez-Hernández, L. Plaja, Comment on 'on the dipole, velocity and acceleration forms in high-order harmonic generation from a single atom or molecule', Journal of Physics B: Atomic, Molecular and Optical Physics 45 (2) (2012) 028001. `doi:10.1088/0953-4075/45/2/028001`.

[75] A. Gordon, F. X. Kärtner, The Ehrenfest Theorem and Quantitative Predictions of HHG based on the Three-step Model, Springer New York, New York, 2007, pp. 183–186. `doi:10.1007/978-0-387-49119-6_23`.

[76] Optical constants of the noble metals, Phys. Rev. B 6 (1972) 4370–4379. `doi:10.1103/PhysRevB.6.4370`.

[77] M. N. Polyanskiy, Refractive index database, [Online; accessed Nov. 19th 2018]. URL `https://refractiveindex.info`

[78] C. Ropers, D. R. Solli, C. P. Schulz, C. Lienau, T. Elsaesser, Localized multiphoton emission of femtosecond electron pulses from metal nanotips, Phys. Rev. Lett. 98 (2007) 043907. `doi:10.1103/PhysRevLett.98.043907`.

[79] H. B. Michaelson, The work function of the elements and its periodicity, Journal of Applied Physics 48 (11) (1977) 4729–4733. `doi:10.1063/1.323539`.