

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/275156822>

# DMTDHF: A full dimensional time-dependent Hartree-Fock program for diatomic molecules in strong laser fields

Article in *Computer Physics Communications* · April 2015

Impact Factor: 3.11 · DOI: 10.1016/j.cpc.2015.04.008

---

READS

71

## 3 authors:



[Bin Zhang](#)

National University of Defense Technology

13 PUBLICATIONS 63 CITATIONS

[SEE PROFILE](#)



[Jianmin Yuan](#)

National University of Defense Technology

233 PUBLICATIONS 1,858 CITATIONS

[SEE PROFILE](#)



[Zengxiu Zhao](#)

National University of Defense Technology

96 PUBLICATIONS 1,356 CITATIONS

[SEE PROFILE](#)



# DMTDHF: A full dimensional time-dependent Hartree–Fock program for diatomic molecules in strong laser fields<sup>☆</sup>



Bin Zhang, Jianmin Yuan, Zengxiu Zhao<sup>\*</sup>

College of Science, National University of Defense Technology, Changsha 410073, China

## ARTICLE INFO

### Article history:

Received 12 November 2014

Received in revised form

30 March 2015

Accepted 7 April 2015

Available online 18 April 2015

### Keywords:

Diatomic molecule

TDHF

Strong field ionization

HHG

## ABSTRACT

We present the DMTDHF package, which investigates the non-perturbative electronic dynamics of diatomic molecules subjected to strong external laser fields by the fully propagated time-dependent Hartree–Fock theory. The package uses the prolate spheroidal coordinates, together with the finite-element method and discrete-variable representation, while short iterative Lanczos algorithm is employed for the time propagation. This package can be applied to diatomic molecules with many electrons, and is implemented to allow easy extensions for additional capabilities.

### Program summary

*Program title:* DMTDHF

*Catalogue identifier:* AEWN\_v1\_0

*Program summary URL:* [http://cpc.cs.qub.ac.uk/summaries/AEWN\\_v1\\_0.html](http://cpc.cs.qub.ac.uk/summaries/AEWN_v1_0.html)

*Program obtainable from:* CPC Program Library, Queen's University, Belfast, N. Ireland

*Licensing provisions:* Standard CPC licence, <http://cpc.cs.qub.ac.uk/licence/licence.html>

*No. of lines in distributed program, including test data, etc.:* 702996

*No. of bytes in distributed program, including test data, etc.:* 7336872

*Distribution format:* tar.gz

*Programming language:* Fortran 2003.

*Computer:* All computers with a Fortran compiler supporting at least Fortran 2003.

*Operating system:* All operating systems with such a compiler. The makefile depends on a Unix-like system and needs modification under Windows.

*Has the code been vectorized or parallelized?:* The program is able to run both in sequential and parallel mode. For parallel running, the number of processors can be arbitrary.

*RAM:* Several GBs, depends on the size of the molecules and the number of DVR basis functions.

*Classification:* 16.1, 16.2, 16.6.

*External routines:* BLAS, LAPACK, FFTW3, MPI

*Nature of problem:* The investigation of molecules interacting with intense laser pulses requires non-perturbative theoretical treatment. It has been evidenced that multiorbital and multipole effects come into play for strong-field physics, while the direct numerical integration of TDSE is computationally prohibitive for systems with more than two electrons. TDHF goes beyond the SAE approach and includes the response to the field of all electrons, which is helpful to resolve the multielectron effects from the collective response of electrons.

*Solution method:* The package uses the prolate spheroidal coordinates, together with the finite-elements method and discrete-variable representation, while short iterative Lanczos algorithm is employed for the time propagation.

<sup>☆</sup> This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (<http://www.sciencedirect.com/science/journal/00104655>).

<sup>\*</sup> Corresponding author.

E-mail address: [zhao.zengxiu@gmail.com](mailto:zhao.zengxiu@gmail.com) (Z. Zhao).

*Restrictions:* Currently, the spin-restricted form of TDHF is implemented, which can be applied to closed-shell molecules. However, the extension to spin-unrestricted form is straight forward.

*Unusual features:* Due to the prolate spheroidal coordinates, the Coulomb potential is treated accurately. The calculations can converge with only a mild number of basis functions and are highly efficient due to the nature of DVR functions in calculation of the two-electron integrals.

*Running time:* The running time depends on the size of the molecules, the number of basis functions, the duration of laser pulses, and the number of processors used. Depending on these factors, it can vary between a few hours and weeks.

© 2015 Elsevier B.V. All rights reserved.

## 1. Introduction

In recent times considerable attention has been paid to the investigation of molecules in intense laser pulses, both experimentally and theoretically [1,2], especially when the recent advances in laser technology have made possible experimental investigation of molecular processes on an ultrashort time scale and under ultrastrong laser intensity [3–5]. Multiphoton excitation, strong-field ionization, and higher-order harmonic generation (HHG) are examples of non-perturbative electronic dynamics [3], which require non-perturbative treatment in return. The direct numerical integration of the time-dependent Schrödinger equation (TDSE) has been successful in explaining many experimental results, but it is computationally prohibitive for systems with more than two electrons [6–8]. Due to the extra degrees of freedom, the response of molecules to strong fields is considerably more complicated than that of atoms, which poses great theoretical and computational challenge. Diatomic molecules are of fundamental importance in this regard and provide an important test-bed. For diatomic molecules, the extra degrees of freedom are the internuclear separation  $R$ , and the alignment (orientation) of the molecular axis with respect to the laser field polarization direction, which can be controlled by pulsed lasers [9]. This program is based on the fixed nuclei approximation and focuses on the dependence of strong-field processes on the orientation effects.

Great difficulty still exists even for the theoretical investigations of diatomic molecules with many electrons. In the frequently used single-active-electron (SAE) approximation, the dynamics of one electron is calculated in the potential of the others in their frozen undistorted field-free orbitals [10,11]. Theoretical models based on SAE approximation are also widely used, such as the molecular tunneling ionization theory (MO-ADK) [12,13], the molecular strong-field approximation (MO-SFA) [or referenced as the Keldysh–Faisal–Reiss (KFR) [14–16] approximation], and the Lewenstein HHG model [17]. However, problems may arise in these numerical and model calculations due to the SAE nature [18–21].

There is clearly a need for a theory which can treat multielectron dynamics self-consistently in strong time-dependent electric fields. One popular choice is the time-dependent density-functional theory (TDDFT) [22–25]. For TDDFT, the advantage is that they are in general less costly in terms of computation while electron correlation is accounted for to some extent. However, the accuracy of TDDFT methods depends critically on the formulation of the exchange–correlation (XC) potential. In general, the XC potentials are memory-dependent, while up to now most TDDFT calculations rely on the adiabatic approximation. Another obvious challenge is the construction of proper functional forms for physical quantities in concern [26]. A reliable alternative is the multiconfiguration time-dependent Hartree–Fock (MCTDHF) theory [27–29], but this method is computationally far more expensive.

At present, the implementation of MCTDHF method is limited to either 1D approximations or very small molecules.

In this paper, we present the DMTHF package, which treats the response of diatomic molecules subjected to intense external laser fields by the fully propagated time-dependent Hartree–Fock (TDHF) theory [30,31], within the fixed nuclei approximation. TDHF is a single determinant theory and may therefore be applied to quite large systems. It goes beyond the SAE approach and includes the response to the field of all electrons. TDHF does not include correlation, but it can be regarded as a good starting point from which one can move to higher levels of theory [32,33]. In strong-field cases, the question of how much and under what conditions correlation beyond the Hartree–Fock model is important still remains discussed. This package is helpful to partially resolve the multielectron effects from the collective response of electrons within the framework of Hartree–Fock.

For the numerical implementation of this package, we use the prolate spheroidal (PS) coordinates [34], which is almost natural choice for two-center systems. Our approach is also based on the discrete-variable representation (DVR) and the finite-element method (FEM) [35]. DVR offers distinct advantages in the representation of local potential operators, while FEM provides more flexibility in the design of numerical grid and increases the sparsity of the kinetics matrix. For the temporal propagation, we use the efficient short iterative Lanczos (SIL) algorithm [36]. For efficient matrix and vector operations we use the basic linear algebra subroutines (BLAS) and the linear algebra package (LAPACK) [37]. It is well known that the calculations of two-electron exchange integrals are the most time-consuming part in TDHF calculation. In this work, the package is also parallelized with the message-passing interface (MPI) to speed up the calculation.

This paper is organized as follows. In Section 2 we present the theoretical background on which the package is built, and the numerical methods are introduced in Section 3. In Section 4 we exhibit an overview of the package structure. The input description is presented in Section 5. The output files are described in Section 6. Section 7 presents the installation and use instructions. In Section 8 we show several test applications of the package. We present our conclusions in Section 9. The command lines used in this paper are listed in the Appendix.

## 2. Theoretical methods

### 2.1. TDHF

Generally, the TDSE is given by  $i\partial\Psi(t)/\partial t = H(t)\Psi(t)$ , where the multi-electron Hamiltonian reads (atomic units are used unless otherwise stated),

$$\hat{H}(t) = \sum_i \hat{h}(i, t) + \sum_{i<j} \hat{g}(i, j). \quad (1)$$

Here the single-electron Hamiltonian

$$\hat{h}(i, t) = -\nabla_i^2/2 - \sum_p Z_p/r_{pi} + v_f(i, t). \quad (2)$$

In the second term, the summation  $p$  runs over the nucleus with charge  $Z_p$  and location  $\mathbf{R}_p$ , and  $r_{pi} = \|\mathbf{r}_i - \mathbf{R}_p\|$ . The field–electron interaction in the dipole approximation is given by  $v_f(i, t) = \mathbf{F}(t) \cdot \mathbf{r}_i$ , where  $\mathbf{F}(t)$  is the laser electric field. The two-body operator,

$$\hat{g}(i, j) = 1/r_{ij}, \quad (3)$$

with  $r_{ij} = \|\mathbf{r}_i - \mathbf{r}_j\|$ .

In TDHF approximation, the time-dependent multi-electron wavefunction is a single Slater determinant [30],

$$\Psi(t) = \hat{\mathbf{A}} \left\{ \prod_{i=1}^{N_e} \psi_i(\mathbf{r}_i, t) \right\} \quad (4)$$

where each one-electron  $\psi_i(\mathbf{r}_i, t)$  stands for a spinorbital, and  $\hat{\mathbf{A}}$  is the antisymmetrizer.

The equations of motion are found with the Lagrange formulation of the time-dependent variation principle which requires a minimization of the action function [28],

$$S[\{\psi_k\}] = \int dt \left\{ \left\langle \Psi \left| \hat{H} - i \frac{\partial}{\partial t} \right| \Psi \right\rangle - \sum_{kl} u_{kl}(t) (\langle \psi_k | \psi_l \rangle - \delta_{kl}) \right\} \quad (5)$$

with respect to the orbitals. The Lagrange multipliers  $u_{kl}(t)$  are introduced to ensure the orthonormality of the orbitals during the temporal evolution. After some algebras, the functional derivatives  $\delta S[\delta\psi_n] = 0$  results in the TDHF equation,

$$\left( \hat{F} - i \frac{\partial}{\partial t} \right) |\psi_n\rangle = \sum_l u_{nl}(t) |\psi_l\rangle \quad (6)$$

where the Fock operator  $\hat{F} = \hat{h} + \hat{J} - \hat{K}$ , with  $\hat{J} = \sum_i \langle \psi_i | \hat{g} | \psi_i \rangle$  and  $\hat{K}$  defined as  $\hat{K} |\psi_n\rangle = \sum_i \langle \psi_i | \hat{g} | \psi_n \rangle |\psi_i\rangle$ . In Eq. (6) the elimination of the Lagrange multipliers led to an explicit form,

$$i \frac{\partial |\psi_n\rangle}{\partial t} = \hat{P} \hat{F}(t) |\psi_n\rangle \quad (7)$$

with the projection operator  $\hat{P} = 1 - \sum_m |\psi_m\rangle \langle \psi_m|$ , which projects on the orthogonal complement of the subspace spanned by the orbitals. Note that the condition  $\langle \psi_m | \frac{\partial}{\partial t} |\psi_n\rangle = 0$  (a unitary transformation) is used here. In this work, for the closed shell molecules, we use the spin-restricted form of Eq. (7), while the spin-unrestricted version is straightforward for open shell molecules.

The orbital  $\psi$  is expanded with a set of basis functions  $\psi_u = \sum_i c_{iu} \phi_i$ , ( $i = 1, 2, \dots, n_b$ ), where  $c_{iu}$  is the expansion coefficient. To calculate the matrix of the Fock operator, we first define the density matrix,

$$D_{ij} = \sum_u^{n_{occ}} c_{iu} c_{ju}^* \quad (8)$$

where the summation runs over the  $n_{occ}$  occupied orbitals. Then, the Fock matrix is calculated as [38]

$$F_{ij} = h_{ij} + \sum_{kl} D_{lk} (2g_{ijkl} - g_{ilkj}) \quad (9)$$

where the one-electron term  $h_{ij} = \langle \phi_i | \hat{h} | \phi_j \rangle$ , and the two-electron term  $g_{ijkl} = \int \phi_i^*(1) \phi_j(1) \hat{g}(1, 2) \phi_k^*(2) \phi_l(2) d\mathbf{r}_1 d\mathbf{r}_2$ . The calculation

$g_{ijkl}$  scales as  $n_b^4$  for a general basis set, and is the most time consuming part in TDHF simulations. With  $D$  and  $F$ , the total energy of the molecule is

$$E = \sum_{ij} D_{ji} (h_{ij} + F_{ij}) + \sum_{p<q} \frac{Z_p Z_q}{R_{pq}}, \quad (10)$$

where the last term is the nuclear repulsion energy, with  $R_{pq} = \|\mathbf{R}_p - \mathbf{R}_q\|$ .

## 2.2. Prolate spheroidal coordinates

If the molecular axis of the diatomic molecule is directed along the  $z$  axis and the nuclei are located on this axis at the positions  $-a$  and  $a$  (the internuclear separation  $R = 2a$ ), then the prolate spheroidal coordinates  $\xi$ ,  $\eta$ , and  $\varphi$  are related to the Cartesian coordinates  $x$ ,  $y$ , and  $z$  as follows [34]:

$$\begin{aligned} x &= a\sqrt{(\xi^2 - 1)(1 - \eta^2)} \cos \varphi, \\ y &= a\sqrt{(\xi^2 - 1)(1 - \eta^2)} \sin \varphi, \\ z &= a\xi\eta, \end{aligned} \quad (11)$$

where  $1 \leq \xi < \infty$  and  $-1 \leq \eta \leq 1$ .

The Coulomb interaction between an electron and two nuclei with charges  $Z_1$  and  $Z_2$  is given by

$$v_n(\xi, \eta) = -\frac{(Z_1 + Z_2)\xi + (Z_2 - Z_1)\eta}{a(\xi^2 - \eta^2)}, \quad (12)$$

which is rendered benign by the volume element  $dV = a^3(\xi^2 - \eta^2)d\xi d\eta d\varphi$ .

The Laplacian operator in these coordinates reads,

$$\begin{aligned} \nabla^2 &= \frac{1}{a^2(\xi^2 - \eta^2)} \left[ \frac{\partial}{\partial \xi} (\xi^2 - 1) \frac{\partial}{\partial \xi} + \frac{\partial}{\partial \eta} (1 - \eta^2) \frac{\partial}{\partial \eta} \right. \\ &\quad \left. + \frac{\xi^2 - \eta^2}{(\xi^2 - 1)(1 - \eta^2)} \frac{\partial^2}{\partial \varphi^2} \right]. \end{aligned} \quad (13)$$

The orbital  $\psi(\xi, \eta, \varphi; t)$  is expanded in the Fourier series with respect to the angular coordinate  $\varphi$ ,

$$\psi(\xi, \eta, \varphi; t) = \sum_{m=-M}^M \phi_m(\xi, \eta; t) \Phi_m(\varphi), \quad (14)$$

where  $\Phi_m(\varphi) = 1/\sqrt{2\pi} e^{im\varphi}$ , and  $M$  is the maximum magnetic number in the expansion. The partial wave  $\phi_m(\xi, \eta; t)$  is further expanded in a product basis of FE-DVR/DVR functions,

$$\phi_m(\xi, \eta; t) = \sum_{ij} c_{ij}^m(t) f_i(\xi) g_j(\eta). \quad (15)$$

Note that the exact eigenfunctions have factors  $(\xi^2 - 1)^{|m|/2} (1 - \eta^2)^{|m|/2}$  which are nonanalytical at  $\xi = 1$  and  $\eta = \pm 1$  (the molecular axis) for odd  $m$  [8,39]. To ensure accurate numerical solutions of the differential equations for both even and odd projections of angular momentum, the same measure is taken just as in the case of hydrogen molecular ion in our previous work [40], which will be illustrated in the following text.

## 2.3. FE-DVR

For the  $\xi$  coordinate, we use the FE-DVR basis functions. The truncated interval  $[1, \xi_{max}]$  (with sufficiently large  $\xi_{max}$ ) is divided into  $n_e$  finite elements with arbitrary boundaries  $\xi^1 = 1 < \xi^2 < \xi^3 < \dots < \xi^{n_e-1} < \xi^{n_e} = \xi_{max}$ . Each FE  $l$  (i.e., in  $[\xi^l, \xi^{l+1}]$ ) is

discretized using the generalized Gauss-quadrature [35] abscissas  $\xi_k^l$  and weights  $w_k^l$ :

$$\begin{aligned} \xi_k^l &= \frac{1}{2}[(\xi^{l+1} - \xi^l)x_k + (\xi^{l+1} + \xi^l)], \\ w_k^l &= \frac{1}{2}(\xi^{l+1} - \xi^l)w_k \end{aligned} \quad (16)$$

where the abscissas and weights  $[\{x_k, w_k\}, k = 1, \dots, n_g]$  are associated with the standard ( $n_g$ ) points: Gauss–Radau quadrature ( $x_{n_g} = 1$ ) for the first element and Gauss–Lobatto quadrature [41] ( $x_1 = -1, x_{n_g} = 1$ ) for the others. The FE-DVR basis functions are

$$f_k^l(\xi) = \begin{cases} \frac{\chi_{n_g}^l(\xi) + \chi_1^{l+1}(\xi)}{\sqrt{w_{n_g}^l + w_1^{l+1}}}, & (k = n_g, \text{ bridge}), \\ \frac{\chi_k^l(\xi)}{\sqrt{w_k^l}}, & (k \neq n_g, \text{ else}), \end{cases} \quad (17)$$

where the interpolation functions for even and odd  $m$  are

$$\chi_k^l(\xi) = \begin{cases} \prod_{t \neq k} \frac{\xi - \xi_t^l}{\xi_k^l - \xi_t^l}, & (m = \text{even}), \\ \sqrt{\frac{\xi^2 - 1}{\xi_k^{l2} - 1}} \prod_{t \neq k} \frac{\xi - \xi_t^l}{\xi_k^l - \xi_t^l}, & (m = \text{odd}), \end{cases} \quad (18)$$

for  $\xi^l \leq \xi \leq \xi^{l+1}$ , and  $\chi_k^l(\xi) = 0$  for  $\xi < \xi^l$  as well as  $\xi > \xi^{l+1}$ . The bridge functions ( $k = n_g$ ) in Eq. (17) is build to ensure communication and continuity between two adjacent element (element  $l$  and  $l + 1$ ). Generally in Eq. (17), the element's superscript  $l$  runs from 1 to  $n_e$ , and the local DVR subscript  $k$  is in the range 2, 3, ...,  $n_g$  ( $k$  include 1 for  $l = 1$ ), with exception that the last DVR of the last FE ( $l = n_e, k = n_g$ ) is removed to ensure that the wavefunction vanishes outside the interval  $[1, \xi_{max}]$ . Finally, the number of FE-DVR functions for the  $\xi$  coordinate is  $n_\xi = n_e n_g - (n_e - 1) - 1 = n_e(n_g - 1)$ .

On the other hand, the variable  $\eta$  (which runs from  $-1$  to  $1$ ) is discretized using a single set of DVR functions,

$$g_j(\eta) = \begin{cases} \frac{1}{\sqrt{w_j^\eta}} \prod_{t \neq j} \frac{\eta - \eta_t}{\eta_j - \eta_t}, & (m = \text{even}) \\ \frac{1}{\sqrt{w_j^\eta}} \sqrt{\frac{1 - \eta^2}{1 - \eta_j^2}} \prod_{t \neq j} \frac{\eta - \eta_t}{\eta_j - \eta_t}, & (m = \text{odd}) \end{cases} \quad (19)$$

where  $[\{\eta_j, w_j^\eta\}, j = 1, \dots, n_\eta]$  are the abscissas and associated weights corresponding to the  $n_\eta$ -points Gauss–Legendre quadrature [41].

From definition, both  $f$  and  $g$  satisfy the orthonormal condition,  $\langle f_k^l | f_{k'}^l \rangle = \delta_{kk'}$ ,  $\langle g_j | g_{j'} \rangle = \delta_{jj'}$ . The total dimension of the basis is thus  $n_b = n_\xi n_\eta (2M + 1)$ . In the following, we use the simplified one-electron basis notation,  $|ijm\rangle = |f_i(\xi)g_j(\eta)\Phi_m(\phi)\rangle$ , and two-electron basis notation,  $|ijklm_1m_2\rangle = |f_i(\xi_1)f_j(\xi_2)g_k(\eta_1)g_l(\eta_2)\Phi_{m_1}(\phi_1)\Phi_{m_2}(\phi_2)\rangle$ .

Any local potential operator is diagonal in the FE-DVR/DVR basis functions, and can be represented by a vector. In the following, we only introduce briefly the calculations of the Kinetics matrix and electron–electron Coulomb interaction integrals.

#### 2.4. Kinetics integration

The matrix of Kinetics operator is calculated as

$$K = \left\langle ijm \left| -\frac{\nabla^2}{2} \right| i'j'm' \right\rangle. \quad (20)$$

As the FE-DVR basis functions are different for even and odd  $m$ , the calculations are performed separately for  $m$  with different parities. Using Eq. (13), and after some algebras, the results for even  $m$  reads [39],

$$\begin{aligned} K &= \frac{a\delta_{ij'}\delta_{mm'}}{2} \int (\xi^2 - 1) \frac{\partial f_i}{\partial \xi} \frac{\partial f_{i'}}{\partial \xi} d\xi + \frac{a\delta_{ii'}\delta_{mm'}}{2} \int (1 - \eta^2) \\ &\times \frac{\partial g_j}{\partial \eta} \frac{\partial g_{j'}}{\partial \eta} d\eta + \frac{a\delta_{ii'}\delta_{jj'}\delta_{mm'}}{2} \frac{(\xi_i^2 - \eta_j^2)m^2}{(\xi_i^2 - 1)(1 - \eta_j^2)}. \end{aligned} \quad (21)$$

For odd  $m$ , due to the additional factors in the FE-DVR basis functions, the expression is more complicated,

$$\begin{aligned} K &= \frac{a\delta_{ij'}\delta_{mm'}}{2} \int (\xi^2 - 1) \left[ (\xi^2 - 1) \frac{\partial f_i}{\partial \xi} \frac{\partial f_{i'}}{\partial \xi} + \xi \left( f_i \frac{\partial f_{i'}}{\partial \xi} \right. \right. \\ &+ \left. \left. \frac{\partial f_i}{\partial \xi} f_{i'} \right) \right] d\xi + \frac{a\delta_{ii'}\delta_{mm'}}{2} \int (1 - \eta^2) \left[ (1 - \eta^2) \right. \\ &\times \left. \frac{\partial g_j}{\partial \eta} \frac{\partial g_{j'}}{\partial \eta} - \eta \left( g_j \frac{\partial g_{j'}}{\partial \eta} + \frac{\partial g_j}{\partial \eta} g_{j'} \right) \right] d\eta + \frac{a\delta_{ii'}\delta_{jj'}\delta_{mm'}}{2} \\ &\times \frac{(\xi_i^2 - \eta_j^2)(m^2 + 1)}{(\xi_i^2 - 1)(1 - \eta_j^2)}. \end{aligned} \quad (22)$$

The matrix elements are diagonal with respect to the magnetic quantum number. Note that in Eq. (22) for odd  $m$ , the functions  $f_i$  and  $g_j$  already take the forms of even  $m$ . The additional factors have been applied in the expression, resulting in several more terms.

In Eqs. (21) and (22), the remaining integrations over  $\xi$  and  $\eta$  can be performed directed with the Gaussian integration method. For example, for  $\eta$ , the integration

$$\int h(\eta) \frac{\partial g_{j_1}}{\partial \eta} \frac{\partial g_{j_2}}{\partial \eta} d\eta = \sum_j h(\eta_j) \frac{\partial g_{j_1}(\eta_j)}{\partial \eta} \frac{\partial g_{j_2}(\eta_j)}{\partial \eta} w_j^\eta, \quad (23)$$

where  $h(\eta)$  is an arbitrary function of  $\eta$ . For  $\xi$ , the integration depends on the type of the basis functions,

$$\begin{aligned} \int h(\xi) \frac{\partial f_k^l}{\partial \xi} \frac{\partial f_{k'}^l}{\partial \xi} d\xi &= \begin{cases} \delta_{ll'} t_{kk'}^l (w_k^l w_{k'}^l)^{-\frac{1}{2}}, & (k \neq n_g, k' \neq n_g) \\ (\delta_{ll'} t_{n_g k'}^l + \delta_{l+1, l'} t_{1 k'}^{l+1}) [(w_{n_g}^l + w_1^{l+1}) w_{k'}^l]^{-\frac{1}{2}}, & (k = n_g, k' \neq n_g) \\ (\delta_{ll'} t_{k n_g}^l + \delta_{l+1, l'} t_{k 1}^{l+1}) [w_k^l (w_{n_g}^l + w_1^{l+1})]^{-\frac{1}{2}}, & (k \neq n_g, k' = n_g) \\ [\delta_{ll'} (t_{n_g n_g}^l + t_{1 1}^{l+1}) + \delta_{l+1, l'} t_{n_g}^l + \delta_{l, l+1} t_{n_g}^l] \\ \times [(w_{n_g}^l + w_1^{l+1}) (w_{n_g}^l + w_1^{l+1})]^{-\frac{1}{2}}, & (k = n_g, k' = n_g) \end{cases} \end{aligned} \quad (24)$$

where

$$t_{k_1 k_2}^l = \sum_k h(\xi_k^l) \frac{\partial \chi_{k_1}^l(\xi_k^l)}{\partial \xi} \frac{\partial \chi_{k_2}^l(\xi_k^l)}{\partial \xi} w_k^l. \quad (25)$$

From Eq. (24), we see that the Kinetics matrix has a quasi-block-diagonal structure with the  $\xi$  coordinate, which increases the sparsity of the Kinetics matrix.

#### 2.5. Two-electron integration

We calculate the electron–electron Coulomb interaction integral,

$$G = \left\langle ijklm_1m_2 \left| \frac{1}{r_{12}} \right| i'j'k'l'm'_1m'_2 \right\rangle, \quad (26)$$

by expanding  $1/r_{12}$  in prolate spherical coordinates through the Neumann expansion [8]

$$\frac{1}{r_{12}} = \frac{1}{a} \sum_{l=0}^{\infty} \sum_{m=-l}^l (-1)^{|m|} (2l+1) \left( \frac{(l-|m|)!}{(l+|m|)!} \right) e^{im(\phi_1-\phi_2)} \times P_l^{|m|}(\xi_{<}) Q_l^{|m|}(\xi_{>}) P_l^{|m|}(\eta_1) P_l^{|m|}(\eta_2) \quad (27)$$

where  $\xi_{>(<)} = \max(\min)(\xi_1, \xi_2)$ . Both the regular  $P_l^{|m|}(\xi)$  and irregular  $Q_l^{|m|}(\xi)$  Legendre functions, which are defined in the region  $(1, +\infty)$ , are involved in the expansion as the “radial” part, while the “angular” part is only related to  $P_l^{|m|}(\eta)$ .

Following the steps in Ref. [8], the integration  $G$  can be expressed as,

$$G = \frac{1}{a} \sum_{l=|m|}^{\infty} (-1)^{|m|} (2l+1) \left( \frac{(l-|m|)!}{(l+|m|)!} \right) \mathcal{J}_{ijkl}^{i'j'k'l'}(l) \quad (28)$$

where  $m = m_1 - m'_1 = m'_2 - m_2$ , which is uniquely determined for a given pair of angular partial waves. The reduced integral reads,

$$\mathcal{J}_{ijkl}^{i'j'k'l'}(l) = \delta_{i'i'} \delta_{j'j} \delta_{k'k} \delta_{l'l} a^6 (\xi_i^2 - \eta_k^2) (\xi_j^2 - \eta_l^2) \times P_l^{|m|}(\eta_l) P_l^{|m|}(\eta_k) \left[ \frac{(-1)^{|m|} (l+|m|)!}{\sqrt{w_\xi^i w_\xi^j} (l-|m|)!} [T^{|m|}]_{ji}^{-1} + P_l^{|m|}(\xi_i) P_l^{|m|}(\xi_j) \frac{Q_l^{|m|}(\xi_{\max})}{P_l^{|m|}(\xi_{\max})} \right] \quad (29)$$

where  $[T^{|m|}]^{-1}$  denotes the inverse of the matrix  $T^{|m|}$ ,

$$T_{ij}^{|m|} = - \int_1^{\xi_{\max}} d\xi f_i(\xi) \left[ \frac{d}{d\xi} (\xi^2 - 1) \frac{d}{d\xi} - l(l+1) - \frac{m^2}{\xi^2 - 1} \right] f_j(\xi). \quad (30)$$

The calculation of  $T_{ij}^{|m|}$  is very similar to the  $\xi$  part in Eq. (20), and for the sake of clarity it will not be presented here. From Eq. (29), we can see that this results in a diagonal representation of the matrix elements over both the  $\xi$  and  $\eta$  coordinates, which considerably simplifies the FE-DVR discretization procedure. In the FE-DVR scheme, the numerical demand for  $G$  is  $n_\xi^2 n_\eta^2 (4M+1)$ . This is far less than  $n_b^4 = n_\xi^4 n_\eta^4 (2M+1)^4$ , which underlies the basis for larger basis functions and bigger simulation box.

## 2.6. Short Iterative Lanczos propagation

We employ the SIL method [36] for the temporal propagation. This method has been used in a previous CPC program [42], which focused on the interaction of a strong laser pulse with atoms. The Lanczos algorithm relies on Krylov subspace techniques that were originally introduced to calculate eigenvalues and eigenvectors of (large) matrices. The construction of the orthonormal Krylov subspace follows a three-term recurrence relation [36]:

$$\begin{aligned} |q_0\rangle &= |\psi(t_0)\rangle \\ \beta_0 |q_1\rangle &= \hat{H} |q_0\rangle - \alpha_0 |q_0\rangle \\ \beta_j |q_{j+1}\rangle &= \hat{H} |q_j\rangle - \alpha_j |q_j\rangle - \beta_{j-1} |q_{j-1}\rangle \end{aligned} \quad (31)$$

where the real numbers  $\alpha$  and  $\beta$  are

$$\begin{aligned} \alpha_j &= \langle q_j | \hat{H} | q_j \rangle, \\ \beta_j &= \langle q_{j+1} | \hat{H} | q_j \rangle. \end{aligned} \quad (32)$$

The Hamilton operator in the subspace  $Q_{N_L+1} [= \{|q_0\rangle, |q_1\rangle, \dots, |q_{N_L}\rangle\}]$ ,  $\hat{H}_{ij}^{(Q)} = \langle q_i | \hat{H} | q_j \rangle$ , is real and tridiagonal.

To perform time propagation, we replace the Hamiltonian in the time evolution operator by its approximation in the Krylov subspace,

$$\begin{aligned} \hat{U}^Q &= \exp(-i\hat{H}^{(Q)} \Delta t) \\ &= \sum_I |Z_I\rangle \exp(-ih_I^{(Q)} \Delta t) \langle Z_I|. \end{aligned} \quad (33)$$

Here,  $|Z_I\rangle$  denotes the eigenvector of  $\hat{H}^{(Q)}$  with the eigenvalue  $h_I^{(Q)}$ , which can be obtained by direct diagonalization of  $\hat{H}^{(Q)}$ .

The approximation for the propagated wavefunction then reads

$$|\psi(t_0 + \Delta t)\rangle \approx \hat{U}^Q |\psi(t_0)\rangle. \quad (34)$$

The approximation gets exact for  $N_L \rightarrow n_b$ . The SIL method is unconditionally stable and the propagation scheme is norm conserving for hermitian Hamiltonian. The propagation error is approximately given by [36]

$$\epsilon \approx \frac{\Delta t^{N_L}}{N_L!} \prod_{i=0}^{N_L} \beta_i, \quad (35)$$

which is used as a tolerance parameter to dynamically adjust the order  $N_L$  to obtain a constant error for each time step.

## 3. Numerical methods

### 3.1. Data types

In the module *comm*, a type *db* is defined for 12-digit accuracy. On all present machines it should specify the type of double precision, and REAL(db) and COMPLEX(db) are actually DOUBLE PRECISION and DOUBLE COMPLEX for most compilers. This symbolic type is kept throughout of coding, which makes the code more flexible and compatible for different target environments. Using single precision is not recommended in this package. For data conversion, we use the expression *CMPLX(a,b,db)* for COMPLEX(db), since *CMPLX(a,b)* returns the default single precision. To convert COMPLEX(db) to REAL(db), the function *REAL* or *AIMAG* is used, which returns the real or imaginary part of a double complex argument.

### 3.2. Calculation of ground states

The ground state is needed to start the TDHF calculation. We calculate the ground state by self-consistently solving the static Hartree–Fock equation,

$$F_{stat} |\psi_n\rangle = \epsilon_n |\psi_n\rangle \quad (36)$$

where the subscript  $n$  runs over the occupied orbitals, with  $\epsilon_n$  and  $|\psi_n\rangle$  the orbital energy and wavefunction. The matrix of static Fock operator  $F_{stat}$  has the same form as the time-dependent one in Eq. (9), if the interaction term  $v_f$  is taken off in the single-electron Hamiltonian equation (2).

This package generates the initial guess orbitals of Eq. (36) in three ways. The first way is to diagonalize the single-electron Hamiltonian  $h$ . The eigenstates are then used as guess orbitals. This method is applicable for simple molecules. However, for complex molecules, this guess is quite rough and the self-consistent calculation may oscillate and yield no convergence. In cases like this, we present another way, which extracts orbitals directly from the outputs of general purpose quantum chemistry programs, such as GAMESS [43]. To this end, we employ the SLIMP package that has been developed in our previous work [44]. Convergence and efficiency can be guaranteed in this method. Besides the

above methods, the program also provides an interface to read the orbitals from saved files.

As an alternative choice, the ground state can also be generated by propagating TDHF equations in imaginary times, which replaces the real time  $t$  with imaginary time  $t' = -it$ . Although implemented in this package, as far as computational cost is considered, this method is less efficient than the static HF method. Nevertheless, this method is applicable when the static methods fails (for example, when the single-electron eigenstates are not adequate and the user is not familiar with other quantum chemistry programs). Furthermore, this serves as a test for the TDHF propagation code.

### 3.3. Laser electric fields

The laser electric field (propagating in the z-direction) is given by

$$\mathbf{F}(t) = \frac{F_0 f(t)}{\sqrt{\epsilon^2 + 1}} [\sin(\omega t + \phi_p) \hat{e}_x + \epsilon \cos(\omega t + \phi_p) \hat{e}_y] \quad (37)$$

where  $F_0$  is the amplitude of laser electric field,  $\omega$  is the carrier frequency,  $\phi_p$  is the carrier-envelope phase,  $\epsilon$  is the ellipticity (the major axis of polarization is along  $\hat{e}_x$ ), and  $f(t)$  is the pulse envelop with  $\max[f(t)] = 1$ . In this paper, the molecular axis is fixed at the z axis, and alignment is achieved by changing the propagation direction of the laser fields. We get a laser propagating in an arbitrary direction  $(\theta, \phi)$  by the following rotation:

$$\begin{cases} F'_x = F_x \cos \theta \cos \phi - F_y \sin \phi \\ F'_y = F_x \cos \theta \sin \phi + F_y \cos \phi \\ F'_z = -F_x \sin \theta. \end{cases} \quad (38)$$

The envelop  $f(t)$  implemented in this package takes the following three forms:

#### 1. A sine-squared envelop

$$f(t) = \sin^2 \left( \frac{\pi t}{\tau} \right), \quad (39)$$

for  $0 < t < \tau$ , and  $f(t) = 0$  for other  $t$ . Here  $\tau$  is the pulse duration.

#### 2. A ramped envelop

$$f(t) = \begin{cases} \sin \left( \frac{\pi t}{2t_{on}} \right), & 0 < t < t_{on} \\ 1, & t_{on} \leq t \leq (t_d - t_{off}) \\ \cos \left( \frac{\pi (t - t_d + t_{off})}{2t_{off}} \right), & (t_d - t_{off}) < t < t_d \end{cases} \quad (40)$$

and  $f(t) = 0$  for other  $t$ . Here  $t_{on}$  and  $t_{off}$  are periods of the rising and declining part, and  $t_{const}$  is the constant part in the middle.  $t_d = t_{on} + t_{const} + t_{off}$  is the total duration of laser pulse.

#### 3. A Gaussian envelop

$$f(t) = \exp \left[ \frac{-2 \ln 2 (t - t_d/2)^2}{\tau^2} \right], \quad (41)$$

for  $0 < t < t_d$ , and  $f(t) = 0$  for other  $t$ . Here  $\tau$  is the full width at half maximum (FWHM), and  $t_d$  is the total pulse duration ( $> \tau$ ).

### 3.4. Absorbing boundary

The absorbing layer is defined by using the radius  $r = a\sqrt{\xi^2 + \eta^2 - 1}$  (the distance measured from the center of axis). Let  $r_{mask}$  be the radius where the absorbing layer starts. An absorbing layer between  $r_{mask}$  and  $r_{max} (= a\xi_{max})$  is used to smoothly bring

down the wavefunction and to prevent the unphysical reflection from the boundary. The absorbing function has the following form:

$$M(r) = \begin{cases} 1 & r \leq r_{mask} \\ \cos^{\frac{1}{8}} \left( \frac{\pi (r - r_{mask})}{2(r_{max} - r_{mask})} \right) & r_{mask} < r \leq r_{max}. \end{cases} \quad (42)$$

The box size has to be big enough so that the physical system is not perturbed by the absorbing boundaries. Note that the wavefunctions are no longer normalized as the molecule slowly gets charged. The absorbed charge can be interpreted as an ionization probability. After the time propagation, we yield the final wavefunction  $\Psi(T)$ . The ionization probability from orbital  $i$  is calculated as

$$p_i = 1 - \langle \psi_i(T) | \psi_i(T) \rangle. \quad (43)$$

The total ionization probability is

$$P = 1 - \langle \Psi(T) | \Psi(T) \rangle = 1 - \prod_i (1 - p_i), \quad (44)$$

which can be approximated as  $P \approx \sum_i p_i$  in the case of small ionizations ( $p_i \ll 1$ ).

### 3.5. Harmonic spectra

The total time-dependent dipole moment

$$\mathbf{D}(t) = -\langle \Psi(t) | \sum_i \mathbf{r}_i | \Psi(t) \rangle \approx \sum_i \mathbf{d}_i(t), \quad (45)$$

where  $\mathbf{d}_i(t) = -\langle \psi_i(t) | \mathbf{r}_i | \psi_i(t) \rangle$  is the dipole moment for orbital  $i$ . Here the minus sign accounts for the negative charge of the electron [45]. Then the emitted harmonic spectra is given by

$$S(\omega) = 4\omega^4 / (6\pi c^3) |\tilde{\mathbf{d}}(\omega)|, \quad (46)$$

where  $c$  is the speed of light.  $\tilde{\mathbf{d}}(\omega) = \int_0^T \mathbf{d}(t) \exp(i\omega t) dt$ , and  $\mathbf{d}(t)$  takes  $\mathbf{d}_i(t)$  or  $\mathbf{D}(t)$  for orbital  $i$  or total orbitals, respectively. The time profiles of harmonics are obtained via a Gabor transform [46],

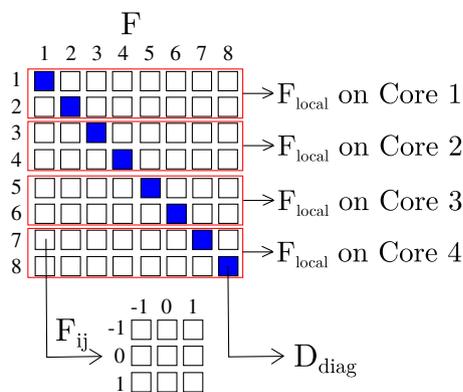
$$\tilde{\mathbf{d}}(\omega, t) = \int_0^T \mathbf{d}(t') \exp(i\omega t') \exp \left[ -\frac{(t' - t)^2}{2\sigma^2} \right] dt' \quad (47)$$

where  $\sigma$  is the width of the Gaussian time window in the Gabor transform.  $\mathbf{d}(t)$  can be differentiated numerically to the first order or second order, yielding a velocity or acceleration form of both Eqs. (46) and (47).

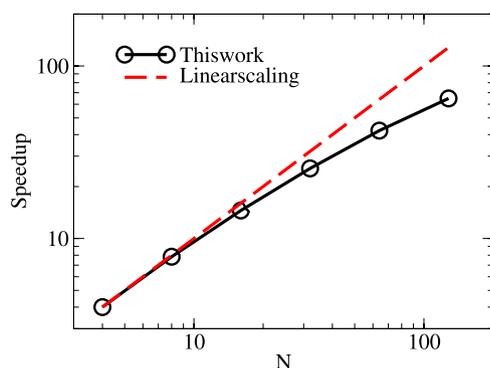
### 3.6. Parallel computation

The Fock matrix ( $F$ ) and density matrix ( $D$ ) are distributed on each CPU core. Let  $N$  stands for the number of CPU cores used. The index  $i$  [in Eq. (9)] is expanded to  $m_i i$ , where  $m_i$  is the  $m$ -quantum number ( $|m_i| \leq M$ ,  $n_m = 2M + 1$ ), and  $i$  is the combined index for coordinates  $\xi$  and  $\eta$  [ $i = 1, 2, \dots, n_g (= n_\xi n_\eta)$ ]. The full matrix is divided into  $n_g \times n_g$  small blocks. Each block ( $F_{ij}$ ) is a matrix of dimension  $n_m \times n_m$ . The matrix is visualized in Fig. 1, where  $N = 4$ ,  $n_g = 8$ , and  $M = 1$ . The distribution of matrix  $F$  is simply dividing the rows ( $n_g$ ) of  $F$  into  $N$  parts, with each part stored locally on one core (the red squares). The distribution is equal if  $n_{mod} \triangleq \text{mod}(n_g, N) = 0$ , while for cases where  $n_{mod} \neq 0$  the residual parts are distributed among the last  $n_{mod}$  cores.

Now let us derive the distribution method of matrix  $D$ , which is slightly different from matrix  $F$ . To differentiate the total index from the expanded index, the total indexes are used in their upper cases in Eq. (9). In Eq. (9), the construction of  $F_{ij}$  relies on two terms on  $D$ : (1) the exchange term  $\sum_{KL} D_{LK} g_{iLKj}$ . From Eq. (29) we see that this summation only contains the terms with  $i = l$ . As a result, the



**Fig. 1.** Illustration of the parallel distribution of the matrix  $F$  and  $D$ . The red squares are the local part stored on each CPU core. The blue filled squares are the diagonal blocks of  $D$ , which are stored on all cores. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 2.** (Color online) Parallel scaling for real-time TDHF propagation of  $H_2$  molecules. The calculations are performed on an IBM BladeCenter H system at NUDT. The speedup is normalized so that at the first data point ( $N = 4$ ) the speedup equals the number of CPU cores.

rows of matrix  $D$  can be divided into  $N$  parts in the same way as  $F$ . (2) the direct term  $\sum_{KL} D_{LK} g_{ijkl}$ . This summation only contains the terms with  $k = l$ . This leads to the dependence of each local  $F$  on all diagonal blocks of  $D$  ( $D_{ii}$ ,  $i = 1, 2, \dots, n_g$ ). As a result, besides the local part of  $D$ , each core also stores the diagonal blocks of  $D$  (the blue blocks).

The orbital wavefunctions are stored fully on each core. Each core calculates the local part of  $F$ , the local part of  $D$  and the diagonal blocks of  $D$ . The SIL propagation method relies mainly on the matrix–vector multiplication  $F|\psi_n\rangle$ , which can be performed in two steps. Firstly, the multiplication is performed locally on each core. Secondly, the orbital wavefunctions are updated with an all-to-all communication of local vector results.

**Fig. 2** presents the parallel scaling for real-time TDHF propagation, in example of the  $H_2$  molecules. We use 8 finite elements in  $\xi$ , with 18 points in each element. The  $\eta$  coordinate is discretized with 18 DVR functions. The angular momentum projections are kept from  $m = -4$  to  $m = 4$ . The total dimension of the basis is thus  $n_b \approx 2.2 \times 10^4$ , which is the typical size of the basis used in this package. The calculations are performed on an IBM BladeCenter H system at NUDT. The result shows that good scaling can be achieved on up to 128 CPU cores, which is usually enough for typical calculations in this package. The scaling begins to level off with more CPU cores. The primary reason for this is that each core needs to store the entire vectors and all-to-all communications are required after local operations of matrix–vector products. For significantly bigger problems, the current parallel algorithms need to be improved to achieve better parallel efficiency.

### 3.7. Active orbitals

For multiorbital molecules, some inner shell orbitals may be more closely bounded than HOMO, which contribute little to the total electronic dynamics. In such cases, the computation cost can be greatly decreased by frozen these orbitals. Three options are provided in this program. (1) Fully propagates all the orbitals. (2) Only the highest several orbitals are propagated, while the inner ones are kept frozen. (3) Only one specific orbital is propagated while all the rest are kept frozen, which is called the single-active orbital (SAO) approximation. Comparison between these methods can be used to investigate the core polarization dynamics in multielectron molecules. The active orbitals can be set by parameters *sae* and *eact*, which are introduced in Section 5.

### 3.8. Restarting a calculation

Sometimes it is necessary to continue a calculation that was not run to the desired completion because of a machine failure or because the number of time steps was set too low. In such cases, a checkpoint file (set by the parameter *fchk*), which contains the last time and wavefunctions at that moment, is generated at regular intervals of *nchk* time steps or at the end of each propagation. This checkpoint file can be used to initialize a continuation. To restart a calculation, one simply need to set the parameter *rtype* to 1 in the input file. Since the computational burdens are usually small for both static HF calculation and imaginary TDHF propagation, restarting is implemented only for the realtime propagation.

## 4. Description of the package

The present package is written as modules in Fortran language. Fully utilizing the spirit of object-oriented method, the package provides a highly convenient and efficient way for the calculation of electronic dynamics of multielectron diatomic molecules subjected to strong laser field though the TDHF theory, while keeping the details embedded inside the modules. To utilize the basic and central functions of the package, the user is never required to make direct contact with the actual data structures. However, the minimal acquaintance with the important types and functions will benefit the user with huge flexibility and potential capability of the package. The modules and driver programs are introduced briefly in the following.

The lower-level modules are:

*alegfunc*: calculates the regular ( $P_l^m$ ) and irregular ( $Q_l^m$ ) associated Legendre functions of arbitrary  $l$  and  $m$ . It was extracted and modified from a previous CPC program [47].

*comm*: the basic constants and data types, and the operations on these types.

*misc*: the auxiliary functions and subroutines used by other modules.

*coordinates*: coordinates transformation between the Prolate Spheroidal, Cartesian, Spherical, and Cylinder coordinates systems.

*integration*: numerical integrations functions and subroutines, including the generation of abscissas and weights of several Gaussian quadratures (Legendre, Radau, and Lobatto).

*interpolation*: the numerical Lagrange and Spline interpolations functions and subroutines.

*files*: file operation functions and subroutines, including the inquiring functions, data input and output functions.

*indexes*: calculates indexes of packed matrix. The general multidimensional matrices are fully packed, while the real symmetric or hermite matrices are packed using the upper triangle parts.

*mpiinfo*: initializes and destroys the MPI environment. Also includes a subroutine for parallel partition of matrices  $D$  and  $F$  among the CPU cores.

*string*: string operations, including transformations between characters and other data types.

*specfunc*: implements several special functions, including the Factorial function, Wigner Rotation function, Spherical harmonic function, Legendre polynomial and its derivative, Lagrange function and its derivative, and so on.

*laser*: calculates the laser electric fields. Both linearly-polarized and elliptically-polarized laser fields are supported. The type *laser\_info* defines the parameters of the electric field. Currently, this module implements the sine-squared, ramped, and Gaussian envelopes introduced in Section 3.3.

*grid*: This module defines the FE-DVR basis functions in Prolate Spheroidal coordinates. The type *fdgrid* defines the parameters of the grid, with a predefined object  $g$ .

*wavefunc*: orbital operation routines, including orbital normalization, projection, and so on. Also includes wavefunction transformations between: (1)  $\psi(\xi, \eta, \phi)$  and the partial waves  $\{\phi_m(\xi, \eta)\}$  in Eq. (14); (2) for partial wave  $\{\phi_m(\xi, \eta)\}$ , the packed forms with the first index or  $\xi \otimes \eta$  the first index.

The higher-level modules are:

*slimp*: extracted and modified from the SLIMP package [44], and is in fact a collection of the following modules: *gto*, *gmo*, *gamess*, *firefly*, *gaussian*, *spec*, and *gwf* in that package. This module calculates the orbital wavefunction directly from the output files of quantum chemistry programs like GAMESS.

*hamiltonian*: constructs the single-electron Hamiltonian in FE-DVR grid, according to Eqs. (21) and (22). Also includes routines for diagonalizing a real symmetric or hermite matrix.

*hartree*: calculates the electron–electron Coulomb integrations in FE-DVR grid, according to Eq. (28). The module *alegfunc* is used here.

*hartreefock*: implements the Spin-restricted Hartree–Fock method. Includes subroutines to generate the initial guess orbitals by diagonalizing the single-electron Hamiltonian or with the SLIMP module.

*silprop*: implements the Short Iterative Lanczos propagation method. The propagation subroutine *sil\_prop* relies on an external subroutine parameter *hmv* which performs the matrix–vector multiplication  $F * \psi$ .

*tdhartreefock*: implements the Spin-restricted Time-dependent Hartree–Fock method. Also includes subroutines to calculate the time-dependent orbital survival probabilities, orbital DMs, and molecular energy.

*hhg*: calculates the harmonic spectra and performs time–frequency analysis by the Gabor transformation.

The driver programs are:

*hf\_driver*: It calls HF initialization routines, sets up the initial guess wavefunctions, and then performs self-consistent calculations until the required precision *etol* is reached.

*tdhf\_driver*: It calls TDHF initialization routines. In realtime propagation, it reads the ground states from a prepared file and propagates the system until the time *tEnd*. If necessary, execution may be terminated on hand by creating an empty file of name *tdhf.exit* in the working directory. In imaginary time propagation, it reads the initial guess states from a prepared file (possibly produced by *hf\_driver*), then propagates the system in imaginary time until the required precision *etol* is reached.

*plot\_driver*: It reads and plots the ground state orbitals generated from *hf\_driver*, or reads a checkpoint file and plots the time-dependent orbitals generated from *tdhf\_driver*. The plotting is

performed on a 2D plane of arbitrary orientation, and the result can be visualized with tools, such as the *gnuplot* program under Linux.

*hhg\_driver*: It calculates the harmonic spectra from the time-dependent DM calculated from *tdhf\_driver*. Time–frequency analysis is performed by the Gabor transformation if the parameter *gabor* is true.

The input parameters and contents of the output files are described in the next two sections.

## 5. Input description

All the inputs are through NAMELIST and are introduced in the following. Any NAMELIST not used for a particular job may be omitted or left in the input file, in which case it is ignored. Most of these variables have default values. In cases where a NAMELIST is used while a variable is not presented, the default value will be used. The input file have to be produced or edited with a text editor.

### 5.1. Namelist input\_grid

This namelist contains parameters of the FE-DVR grid, which are defined in module *grid*:

- (1) *zc1*, *zc2*: charges of the two nuclei, which are located on the z-axis with  $z_1 = -a$  and  $z_2 = a$ , respectively. Default:  $zc1 = 1, zc2 = 1$ .
- (2) *ic*: molecular charge. The default value is 0, indicating a neutral molecule. Note that in the current implementation of spin-restricted form,  $zc1 + zc2 - ic$  must be an even number.
- (3) *nx*: number of Gauss–Radau/Lobatto integration points in each finite element for the  $\xi$  coordinate. Default is 10.
- (4) *ny*: number of Gauss–Legendre integration points for the  $\eta$  coordinate. Default is 10.
- (5) *np*: dimension in the  $\varphi$  coordinate. Default is 36.
- (6) *mmax*: maximum  $m$  quantum number in the orbital expansion of Eq. (14). Default is 0.
- (7) *a*: half bond length, in unit of Bohr radius. Default is 1.0.
- (8) *rmax*: radius of the simulation box ( $r_{max}$ ), in unit of Bohr radius. The maximum of  $\xi$  is thus  $\xi_{max} = r_{max}/a$ . Default is 30.0.
- (9) *rmask*: position where the absorbing boundary starts, in unit of Bohr radius. Default is 20.0.
- (10) *ntype*: number of different types of finite elements for  $\xi$ . The maximum of *ntype* is 10. Default is 2.
- (11) *nt*: number of each type of finite elements for  $\xi$ . Default: {1, 3}.
- (12) *st*: length of each type of finite elements for  $\xi$ . The number of elements in *nt* and *st* is set by *ntype*. For *st*, the last parameter is omitted. Default: {1.0, 5.0}.

The parameters *ntype*, *nt*, and *st* are used to set the finite elements for the  $\xi$  coordinates. The default values are used in the following explanation. Here, *ntype* = 2 means that we have two types of finite elements. *nt* = {1, 3} means that there are 4 finite elements in total, with 1 finite element of type 1 and 3 finite elements of type 2. *st* = {1.0, 5.0} means that the length of finite elements of type 1 is 1.0. Since the summation of all the lengths should be equal to  $(\xi_{max} - 1.0)$  (here 1.0 is the starting value for  $\xi$ ), the last value in *st* (5.0) is not referenced. Each finite element of type 2 spans the rest of space equally. If  $\xi_{max} = 30.0$ , then the length of finite elements of type 2 is  $(30.0 - 1.0 - 1 \times 1.0)/3 \approx 9.33$ .

### 5.2. Namelist input\_laser

This namelist contains parameters of the laser pulse, which are defined in module *laser*.

- (1) *llnten*: intensity of the laser field, in unit of  $10^{14}$  W/cm<sup>2</sup>. Default is 1.0.
- (2) *w*: carrier frequency  $\omega$  of the laser field, in unit of hartree. Default is 0.05695 (corresponding to a carrier wavelength of 800 nm).
- (3) *cep*: carrier envelop phase of the laser field [ $\phi_p$  in Eq. (37)], in unit of degree. Default is 0.
- (4) (*t1*, *t2*): pulse parameters, which will be explained in the following. *t1* is in unit of optical cycle  $T = (2\pi/w)$ . Default:  $t1 = 3.0, t2 = 1.0$ .
- (5) *lEnvelop*: envelop of the laser field. If *lEnvelop* = 0, the pulse has a sine-squared envelop with a pulse duration  $\tau$  [in Eq. (39)] of *t1*. If *lEnvelop* = 1, the pulse has a ramped envelop with constant middle cycles  $t_{const}$  of *t1* and turning on (off) cycles  $t_{on}(t_{off})$  of *t2*, in Eq. (40). If *lEnvelop* = 2, the pulse has a Gaussian envelop with a FWHM  $\tau$  of *t1* and a total pulse duration  $t_d$  of *t2* (in unit of *t1*), in Eq. (41). Default is 0.
- (6) *elp*: ellipticity of the laser field [ $\epsilon$  in Eq. (37)]. Default: 0.0.
- (7) (*theta*, *phi*): the laser field polarization directions, in unit of degree. Default: *theta* = 90.0, *phi* = 0.0, indicating a field polarization direction along the z axis.
- (8) *eOn*: the time when the laser field is turned on, in unit of a.u.. Default is 0.0.

### 5.3. Namelist input\_hf

This namelist is defined in module *hartreefock*, which contains parameters used by the programs *hf\_driver*:

- (1) *guess*: method to generate the initial guess orbitals. If *guess* = 0, the eigenstates of the single-electron Hamiltonian are used as guess orbitals. This guess is quite rough for complex molecules and the calculation may oscillate and yield no convergence. If *guess* = 1, the orbitals from GAMESS, Firefly or GAUSSIAN calculations are read and evaluated. Convergence and efficiency can be guaranteed in this method. If *guess* = 2, the orbitals are read from a prepared file. Default is 0.
- (2) *etol*: the absolute convergence criterion of total molecular energy in the self-consistent calculation, in unit of Hartree. Default is  $10^{-12}$ .
- (3) *maxiter*: the maximum iteration steps for the self-consistent calculation. Default is 100.
- (4) *fgs*: file to write the calculated ground state orbitals. Default: *hfgs.dat*.
- (5) *fguess*: If *guess* = 2, this file contains the initial guess orbitals. Else, writes the initial guess orbitals to this file. Default: *hfgsguess.dat*.
- (6) *fmol*: GAMESS, Firefly or GAUSSIAN output file, used only when *guess* = 1.

### 5.4. Namelist input\_tdhf

This namelist is defined in module *tdhartreefock*, which contains parameters used by the program *tdhf\_driver*:

- (1) *runType*: run type of TDHF calculation. If *runType* = -1, imaginary propagation is performed, with the initial guess orbitals contained in file *fguess*. If *runType* = 0, realtime propagation is performed, with the ground states orbitals contained in file *fgs*. If *runType* = 1, realtime propagation from a check point is performed, with the checkpoint file *fchk*. Default is 0.

- (2) *sae*, *eact*: specifies the active orbitals in the realtime propagation. If *sae* = 0, all orbitals are active. If *sae* = 1, only the orbitals from *eact* to *nocc* are active, while the inner orbitals are kept frozen. If *sae* = 2, all orbitals are frozen except the one specified by *eact*. Default: *sae* = 0, *eact* = 1.
- (3) *ckstep*: number of propagation steps to update the checkpoint file. Default is 1000.
- (4) *fchk*: contains the checkpoint data, including the current time *t*, time step *dt*, and the time-dependent orbitals  $\psi(t)$ . If *runType* = 1, this file is first used as input to restore the propagation status to the moment at which this checkpoint is created. If *runType*  $\neq$  -1, checkpoint data is exported to this file every *ckstep* time steps. Default: *check.dat*.
- (5) (*tZero*, *tEnd*): the initial and maximum time for the realtime propagation, in unit of a.u.. *tZero* is omitted if *runType* = 1. Default: *tZero* = 0.0, *tEnd* = 400.d0.
- (6) *tStep*: propagation time step, in unit of a.u.. Default is 0.01.

### 5.5. Namelist input\_plot

This namelist contains parameters used by the program *plot\_driver*:

- (1) *fwf*: specifies the file that contains the orbitals. Default: *hfgs.dat*.
- (2) *wfityp*: wavefunction type. If *wfityp* = 0, plots the ground state orbital generated from *hf\_driver*. If *wfityp* = 1, plots the time-dependent orbital generated from *tdhf\_driver*. Default is 0.
- (3) *nmo*: the orbital to be calculated. Default is 1.
- (4) *fout*: file to write the results. Default: *plot.dat*.
- (5) (*alpha*, *beta*): rotation angles of the plotting plane, in unit of degree. It denotes the direction of the norm vector of the 2D plotting plane. Defaults are (0.0, 0.0), which means that the plotting plane lies on the x-y plane.
- (6) (*len\_x*, *len\_y*): sizes of the plotting plane, in unit of a.u. Defaults are (10.0, 10.0).
- (7) (*n\_x*, *n\_y*): number of points of the plotting plane. Defaults are (100, 100).
- (8) *modulus*: exports the modulus square of the result if this is true. If this is false, the real or imaginary part of the result is exported according to the parameter *rpart*. Default is *true*.
- (9) *rpart*: this is used only when *modulus* = *false*. If this is true, the real part of the result is exported; else, the imaginary part is exported. Default is *true*.
- (10) *lempty*: if this is true, writes extra empty lines that can be used by the *splot* command in *gnuplot* program. Set *lempty* = *false*. for visualization programs that require no empty lines. Default is *true*.

### 5.6. Namelist input\_hhg

This namelist contains parameters used by the program *hhg\_driver*:

- (1) *fdip*: file to read the time-dependent DM. Default: *dipole.dat*.
- (2) *fhhg*: file to write the harmonic spectra. Default *hhg.dat*.
- (3) *npmax*: the maximum time to be used, in unit of *T*. *npmax* should be larger than the duration of laser pulse. To achieve higher precision for the harmonic spectra, larger *npmax* should be used. Default is 10.
- (4) *gauge*: calculation form. If this is 0, the DM is used directly to calculate the harmonics in the length form. If this is 1(2), the DM is differentiated numerically to the first(second) order, and the velocity(acceleration) form will be employed. Default is 2.
- (5) (*theta*, *phi*): polarization direction of the harmonic spectra to be calculated, in unit of degree. Defaults are (0.0, 0.0) (along the z-axis).

- (6) *gabor*: performs the Gabor analysis if this is true. Default is *false*.
- (7) *tau*: full width of the time window in Gabor analysis, in unit of  $T$ . This is used only when *gabor* is true. Default is 0.2.
- (8) *ntsep*: time steps in Gabor analysis, in unit of  $tStep$ . This is used only when *gabor* is true. Default is 10.
- (9) *lempty*: the same as that in namelist *input\_plot*. Default is *true*.

## 6. Output description

The code produces a number of output files containing various pieces of information. The names of some of the output files can be adjusted using input variables as listed in Section 5, so that those files are here denoted by the default names given there.

*gauleg.dat*, *gaurad.dat*, *gaulob.dat*, *ggauxi.dat*: These files are produced in *init\_grid* in the execution of *hf\_driver*, *tdhf\_driver*, and *plot\_driver*. The purpose is to present and check the abscissas and weights of the elementary Gaussian quadratures: Gaussian–Legendre (*gauleg.dat*), Gaussian–Radau (*gaurad.dat*), Gaussian–Lobatto (*gaulob.dat*), and the generalized Gaussian quadrature for the  $\xi$  coordinates (*ggauxi.dat*). The numbers given in each line are the number, abscissa, and weight.

*mask.dat*: This file is also produced in *init\_grid* in the execution of *hf\_driver*, *tdhf\_driver*, and *plot\_driver*. The purpose is to give a quick impression of the absorbing boundary. The numbers in each line are the radius (a.u.) and the value of masking function.

*hfgs.dat*: This file is produced in the execution of *hf\_driver* or the imaginary propagation of *tdhf\_driver*. It contains the ground state orbital wavefunctions. For each orbital, the numbers in the first line are the orbital number, the  $m$ -quantum number, and the orbital energy (Hartree), and the rest lines are the orbital coefficients. It is used for inputting in the *plot\_driver* or the realtime propagation of *tdhf\_driver*.

*hfsguess.dat*: This file is produced only in the execution of *hf\_driver*. It contains the initial guess wavefunctions of the ground state, where the format of contents is the same as the file *hfgs.dat*. This file is used for inputting in the imaginary propagation of *tdhf\_driver*.

*check.dat*: This file is produced only in the realtime propagation of *tdhf\_driver*, at regular intervals of *nchk* time steps or at the end of propagation. The first number is the current time (a.u.), and the second number is the time step (a.u.). The rest lines are the complete set of wavefunctions. Because this leads to large storage requirements, only the last such file in a run is kept. It can be used for restarting the calculation or for inputting in the *plot\_driver*.

*elaser.dat*: This file is produced only in the realtime propagation of *tdhf\_driver*. The numbers in each lines are the time and the  $x$ ,  $y$ , and  $z$  components of laser electric field, all in units of a.u.

*td\_energy.dat*: This file is produced only in the realtime propagation of *tdhf\_driver*. The numbers in each lines are the time and the total molecular energy, all in units of a.u.

*normX.dat*, *dipX.dat*: These files are produced only in the realtime propagation of *tdhf\_driver*. Here  $X = 0, 1, 2, \dots, n_{occ}$ , where 0 stands for the total molecule, and the others stand for the  $n_{occ}$  occupied orbitals. For *normX.dat*, the numbers in each line are the time and the normalization of wavefunction. For *dipX.dat*, the numbers in each line are the time and the  $x$ ,  $y$ , and  $z$  components of dipole moment. All numbers are in units of a.u. *dipX.dat* can be used for inputting in the *hhg\_driver*.

*im\_energy.dat*, *im\_rtol.dat*: These files are produced only in the imaginary propagation of *tdhf\_driver* and its purpose is to give a quick impression of the convergence behavior. The numbers

in each line are the iteration count, the molecule energy (*im\_energy.dat*) or its relative error (*im\_rtol.dat*), in units of a.u.

*plot.dat*: This file is produced in *plot\_driver*. The numbers in each line are the first and second coordinates of the plotting plane, the plotting result (which depends on the parameters *modulus* and *rpart*). If *lempty* is true, an empty line is put between different values in the first coordinate, and this file can be visualized with the *spot* command in *gnuplot*.

*hhg.dat*, *gb\_hhg.dat*: These files are produced in *hhg\_driver*. *gb\_hhg.dat* is produced only when *gabor* is true. For *hhg.dat*, the numbers in each line are the harmonic order and the spectral intensity (a.u.). For *gb\_hhg.dat*, the numbers are the time, the harmonic order and the spectral intensity (a.u.) from Gabor transformation.

## 7. Installation and use instructions

DMTDHF is distributed as a source code, and needs compiling for the target environment. Before compilation, several instructions should be noted in the following.

For dynamic arrays, this package uses the *allocatable* statement, rather than the *pointer* statement. Allocatable arrays can result in more efficient code due to the contiguous memory. Most importantly, they are automatically deallocated upon exit of the program, avoiding a memory leak. However, the use of allocatable statement in derived types requires a fortran compiler supporting at least Fortran 2003. To compile this program with compilers which only support older Fortran standards (for example Fortran 90 and 95), one has to replace the *allocatable* statement by the *pointer* statement in derived types, together with the replacement of *ALLOCATED* by *ASSOCIATED* in destroying routines.

For efficient matrix and vector operations, this package uses the BLAS and LAPACK libraries. The FFTWS library is employed in the calculation of harmonic spectra. These external libraries should be installed in most scientific computing centers; if not, the source code can be obtained from [www.netlib.org](http://www.netlib.org) or [www.fftw.org](http://www.fftw.org) and compile the libraries yourself. This should work smoothly if the instructions are followed.

The compiler name (FC), options (FCFLAGS) and library names (LIB) in the makefile should be adapted according to compilers used. Execution of the code under MPI will require consulting the local documentation of the target environment.

Compile the package by the command,

*make* or *make all*

which will build the following driver programs: *hf\_driver*, *plot\_driver*, *hhg\_driver*, and *tdhf\_driver*. To run these programs, the user first has to prepare the corresponding input files, and then use the command for the first three programs,

*program* (*filename*)

or the command for *tdhf\_driver* in parallel running (in example of the MPI distributed system),

*mpirun -np N tdfh\_driver* (*filename*)

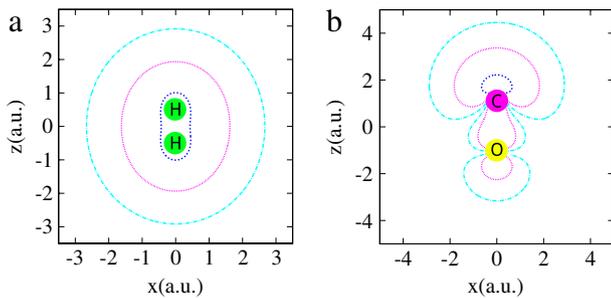
where  $N$  is the number of CPU cores used. If the parameter *filename* is presented, the *program* will use this file as input. If it is not presented, the *program* will search the default file *input* in the current directory.

Input files of several examples are located in the subdirectories of the *tests* directory. The detailed command lines for these examples are listed in the [Appendix](#), for the sake of clarity. The next section presents the calculation results. To verify correct executions, the contents of the generated files can be compared with benchmark results (in the directory *bench* of each example). Agreement should be reached within a reasonable number of digits. If a failure is encountered, please refer to the corresponding log file of the driver program for additional information.

**Table 1**

Total molecular energies from static HF calculations, compared with converged values from literature [48].

	$n_\xi \times n_\eta$	E (a.u.)
$H_2(R = 1.4 \text{ a.u.})$	$36 \times 8$	-1.13353645843
	$44 \times 10$	-1.13362707420
	$52 \times 12$	-1.13362953331
	$68 \times 14$	-1.13362957128
	Ref. [48]	-1.13362957135
$CO(R = 2.132 \text{ a.u.})$	$36 \times 8$	-112.362569945
	$44 \times 10$	-112.773075401
	$52 \times 12$	-112.788779976
	$68 \times 14$	-112.787875463
	Ref. [48]	-112.790906405
Ref. [48]	$193 \times 169$	-112.7909060



**Fig. 3.** (Color online) Visualization of the electron density: (a)  $1\sigma_g$  of  $H_2$ ; (b)  $5\sigma$  of CO. From inside, the contour lines are  $10^{-1}$ ,  $10^{-2}$ , and  $10^{-3}$ , respectively.

## 8. Applications

### 8.1. HF calculations

The ground states of the  $H_2$  and CO molecules are calculated using this package. The internuclear distance  $R$  of these molecules are fixed at experimental equilibriums, which are taken from the NIST database [49]. In Table 1, the total molecular energies are compared with converged values from literature [48], which employed the finite-difference method. Satisfactory precision is achieved for the relatively small basis set we have used. Using the same level of theory, the values given in NIST (with the aug-cc-pVQZ basis) are  $-1.133509$  a.u. for  $H_2$  and  $-112.790812$  a.u. for CO [49], which are very close to the results in this work. Convergence has

been achieved within the current implementation under the frame of Hartree–Fock theory. Due to the neglect of correlation in the Hartree–Fock theory, these results deviates mildly from more accurate correlation-corrected theories.

As an example, the electron density of the HOMO of  $H_2$  ( $1\sigma_g$ ) and CO ( $5\sigma$ ) molecules are visualized on the  $x$ - $z$  plane in Fig. 3, with the *gnuplot* program under Linux. For the homonuclear molecule  $H_2$ , the orbital is inversion symmetric. For CO, which is a heteronuclear molecule, the electron tends to populate the C-end rather than the O-end, which leads to the orientation-dependent ionization and harmonic generation.

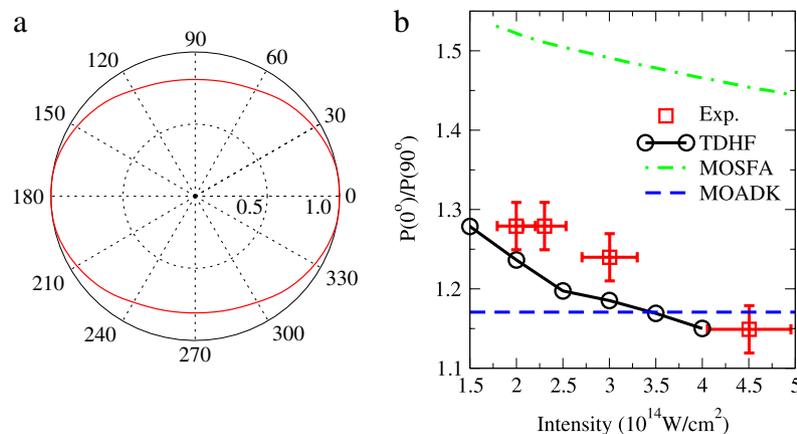
### 8.2. $H_2$ molecules

With the HF ground state calculated in the previous section, the  $H_2$  molecule is propagated in time with TDHF theory. Alignment-dependent ionization probabilities are calculated by changing the alignment angle between the molecular axis and the direction of linearly-polarized laser field. Fig. 4(a) presents the results at the intensity of  $2 \times 10^{14} \text{ W/cm}^2$ . The ionization probability at  $0^\circ$  is larger than  $90^\circ$ . To compare with experiment [50], the ratio of the ionization probabilities  $p(0^\circ)/p(90^\circ)$  is presented in Fig. 4(b), at various laser intensities. The results from MOADK and MOSFA theories are also given for comparison. The result from this work agrees qualitatively with the experiment, yielding an decreasing ratio with increasing intensity. Both models fail to predict the experiment: the ratio is independent of intensity for MOADK results, while an apparent deviation is observed for MOSFA calculations.

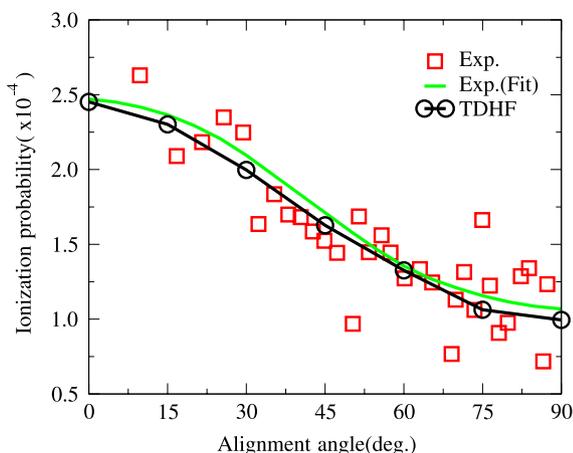
### 8.3. CO molecules

Previous example is on  $H_2$  molecule, which has only one orbital. Now we apply the program to the more complicated CO molecule, with all seven orbitals kept active in the propagation. To compare with experiment [52], the laser intensity of  $0.6 \times 10^{14} \text{ W/cm}^2$  is used, and the results are presented in Fig. 5. For better visualization, the fitted curve of the experimental data is plotted in solid line. Our TDHF results are in good agreement with experiment.

For polar molecules, the electron dynamics are different for parallel ( $0^\circ$ ) and anti-parallel ( $180^\circ$ ) orientations. In our previous works, with few-cycle laser fields, this program has been successfully applied to investigate the orientation-dependent ionization [31] and HHG dynamics [53] of CO molecules.



**Fig. 4.** (Color online) (a) Ionization probability of  $H_2$  versus the alignment angle in linearly polarized, 800 nm laser pulses at  $2 \times 10^{14} \text{ W/cm}^2$ . (b) Ratio of the ionization probabilities  $p(0^\circ)/p(90^\circ)$  as a function laser intensity for theory and experiment. Solid line: TDHF from this work. Dashed line: MOADK [51]. Dotted line: MOSFA [50]. The experimental data is taken from Ref. [50].



**Fig. 5.** (Color online) Ionization probability of CO versus the alignment angle in linearly polarized, 800 nm laser pulses at  $0.6 \times 10^{14}$  W/cm<sup>2</sup>. The TDHF result (circle-marked line) is calculated by fully propagating all the orbitals. The experiment data (square) are taken from [52] (matched to the theory at 0°). The fitted curve of the experimental data is shown in solid line.

## 9. Conclusion

In this paper, we present the DMTDHF package, which can be used to investigate the non-perturbative electronic dynamics of diatomic molecules subjected to intense external laser fields by the fully propagated time-dependent Hartree–Fock theory. We have present simple applications that utilizes the basic functions of the package, while the package can be applied to diatomic molecules with many electrons. Due to the methods used, the package is accurate and efficient, and has been optimized for parallel computations in moderate cases. However, for cases which require relatively bigger simulation boxes and larger basis functions, such as with more intense laser fields or lower carrier frequencies, more CPUs should be used to speed up the calculations. Extensions towards more efficient parallel algorithms are definitely necessary and will be our future work.

## Acknowledgments

This work is supported by the National Basic Research Program of China (973 Program) under Grant No. 2013CB922203, the NSF of China (Grant No. 11374366) and the Major Research plan of NSF of China (Grant No. 91121017).

## Appendix. Command lines

Assume that the current working directory is ‘DMTDHF/tests’. Input files of the examples are located in the subdirectories of this directory. The commands take the H<sub>2</sub> molecules as example, while those on CO molecules are similar to H<sub>2</sub>. To verify correct execution, the contents of the generated files can be compared with benchmark results (in the subdirectory bench) and should agree within a reasonable number of digits.

### (1) Static HF calculations

Change to the directory ‘h2.hf’ and perform the static HF calculation by typing

```
../bin/hf_driver > hf.log
```

The initial guess states and ground states will be written to files ‘hfgsguess.dat’ and ‘hfgs.dat’, respectively. For H<sub>2</sub> the parameter *guess* = 0, the single-electron Hamiltonian is diagonalized to generate the initial guess states. For CO *guess* = 1, the calculation relies on the external file ‘co\_accq’, which was generated by GAMESS with acc-pvqz basis set. The static calculation takes a few seconds on a modern PC.

### (2) Plotting HF ground state orbitals

To run this example, the user first has to generate the HF ground states ‘hfgs.dat’ by running example in (1). Change to the directory ‘h2.plot’ and plot orbital wavefunction by typing

```
../bin/plot_driver > plot.log
```

which will generate the file ‘plot.dat’. Visualize the result with the gnuplot program using the prepared script ‘wfmap’ as:

```
gnuplot wfmap
```

which will generate the file ‘plot.eps’ [Fig. 3(a)].

### (3) Imaginary time TDHF calculations

To run this example, the user first has to generate the initial guess states ‘hfgsguess.dat’ by running example in (1). Change to the directory ‘h2.improp’ and perform the imaginary time TDHF calculation in sequential running by typing

```
../bin/tdhf_driver > tdhf.log
```

or in parallel running by typing

```
mpirun -np N ../bin/tdhf_driver > tdhf.log
```

Here *N* is the number of processors used. The imaginary propagation may takes several minutes.

### (4) Real time TDHF propagations

To run this example, the user first has to generate the HF ground states ‘hfgs.dat’ by running example in (1). Change to the directory ‘h2.tdprop’ and perform the real time TDHF calculation by the same commands as those in example (3).

Here the default value of *theta* is 90.0, indicating a linearly polarized field along the *z* axis (propagating in the *x* axis) and an alignment angle of 0.0. At this parallel alignment, for sequential running, the maximum of *m*-quantum number *mmax* is 0 for H<sub>2</sub> and the execution takes about several minutes, while *mmax* is 1 for CO and the execution takes about several hours. To perform calculation at other alignments [for example *theta* = 0.0 (perpendicular alignment)], *mmax* should takes larger values (typically 4 or larger), the calculation is almost forbidden in sequential running and parallel running is recommended.

### (5) Calculating harmonic spectra

Here we take the CO molecules for example. To run this example, the user first has to produce the time-dependent DM ‘dip7.dat’ by running example in (4). Change to the directory ‘co.hhg’ and calculate the harmonic spectra by typing

```
../bin/hhg_driver > hhg.log
```

The harmonic spectral is in ‘hhg.dat’ and the time–frequency analysis is in ‘gb\_hhg.dat’. Note that the projection direction of harmonics (*theta*, *phi*) in namelist *input\_hhg* is different from the field polarization direction in namelist *input\_laser*.

## References

- [1] T. Brabec, F. Krausz, Intense few-cycle laser fields: *Frontiers of nonlinear optics*, *Rev. Modern Phys.* 72 (2) (2000) 545–591.
- [2] J.H. Posthumus, The dynamics of small molecules in intense laser fields, *Rep. Progr. Phys.* 67 (5) (2004) 623–665. <http://dx.doi.org/10.1088/0034-4885/67/5/R01>.
- [3] F. Krausz, Attosecond physics, *Rev. Modern Phys.* 81 (1) (2009) 163–234. <http://dx.doi.org/10.1103/RevModPhys.81.163>. URL <http://link.aps.org/doi/10.1103/RevModPhys.81.163>.
- [4] M.F. Kling, M.J.J. Vrakking, Attosecond electron dynamics, *Annu. Rev. Phys. Chem.* 59 (November 2007) (2008) 463–492. <http://dx.doi.org/10.1146/annurev.physchem.59.032607.093532>. URL <http://www.ncbi.nlm.nih.gov/pubmed/18031218>.
- [5] M. Protopapas, C.H. Keitel, P.L. Knight, Atomic physics with super-high intensity lasers, *Rep. Progr. Phys.* 60 (1997) 389–486.
- [6] S.X. Hu, Optimizing the FEDVR-TDCC code for exploring the quantum dynamics of two-electron systems in intense laser pulses, *Phys. Rev. E* 81 (5) (2010) 1–7. <http://dx.doi.org/10.1103/PhysRevE.81.056705>. URL <http://link.aps.org/doi/10.1103/PhysRevE.81.056705>.
- [7] Z. Zhang, L.-y. Peng, M.-h. Xu, A.F. Starace, T. Morishita, Q. Gong, Two-photon double ionization of helium: Evolution of the joint angular distribution with photon energy and two-electron energy sharing, *Phys. Rev. A* 84 (4) (2011) 043409. <http://dx.doi.org/10.1103/PhysRevA.84.043409>. URL <http://link.aps.org/doi/10.1103/PhysRevA.84.043409>.

- [8] X. Guan, K. Bartschat, B.I. Schneider, Breakup of the aligned  $H_2$  molecule by xuv laser pulses: A time-dependent treatment in prolate spheroidal coordinates, *Phys. Rev. A* 83 (4) (2011) 043403. <http://dx.doi.org/10.1103/PhysRevA.83.043403>. URL <http://link.aps.org/doi/10.1103/PhysRevA.83.043403>.
- [9] H. Stapelfeldt, Colloquium: Aligning molecules with strong laser pulses, *Rev. Modern Phys.* 75 (2) (2003) 543–557. <http://dx.doi.org/10.1103/RevModPhys.75.543>. URL <http://link.aps.org/doi/10.1103/RevModPhys.75.543>.
- [10] M. Awasthi, Y.V. Vanne, A. Saenz, P. Declava, Single-active-electron approximation for describing molecules in ultrashort laser pulses and its application to molecular hydrogen, *Phys. Rev. A* 77 (6) (2008) 063403. <http://dx.doi.org/10.1103/PhysRevA.77.063403>. URL <http://link.aps.org/doi/10.1103/PhysRevA.77.063403>.
- [11] M. Abu-samha, L.B. Madsen, Single-active-electron potentials for molecules in intense laser fields, *Phys. Rev. A* 81 (3) (2010) 033416. <http://dx.doi.org/10.1103/PhysRevA.81.033416>. URL <http://link.aps.org/doi/10.1103/PhysRevA.81.033416>.
- [12] X.M. Tong, Z.X. Zhao, C.D. Lin, Theory of molecular tunneling ionization, *Phys. Rev. A* 66 (3) (2002) 033402. <http://dx.doi.org/10.1103/PhysRevA.66.033402>. URL <http://link.aps.org/doi/10.1103/PhysRevA.66.033402>.
- [13] T. Kjeldsen, C. Bisgaard, L. Madsen, H. Stapelfeldt, Influence of molecular symmetry on strong-field ionization: Studies on ethylene, benzene, fluorobenzene, and chlorofluorobenzene, *Phys. Rev. A* 71 (1) (2005) 013418. <http://dx.doi.org/10.1103/PhysRevA.71.013418>. URL <http://link.aps.org/doi/10.1103/PhysRevA.71.013418>.
- [14] L.V. Keldysh, Ionization in the field of a strong electromagnetic wave, *Sov. Phys. JETP* 20 (5) (1965) 1307. URL [http://jetp.ac.ru/cgi-bin/dn/e\\_020\\_05\\_1307.pdf](http://jetp.ac.ru/cgi-bin/dn/e_020_05_1307.pdf).
- [15] F.H.M. Faisal, Multiple absorption of laser photons by atoms, *J. Phys. B* 6 (4) (1973) L89–L92. <http://dx.doi.org/10.1088/0022-3700/6/4/011>.
- [16] H.R. Reiss, Effect of an intense electromagnetic field on a weakly bound system, *Phys. Rev. A* 22 (5) (1980) 1786.
- [17] M. Lewenstein, P. Balcou, M. Ivanov, A. L'Huillier, P. Corkum, Theory of high-harmonic generation by low-frequency laser fields, *Phys. Rev. A* 49 (3) (1994) 2117–2132. <http://dx.doi.org/10.1103/PhysRevA.49.2117>. URL <http://link.aps.org/doi/10.1103/PhysRevA.49.2117>.
- [18] Z. Zhao, J. Yuan, T. Brabec, Multielectron signatures in the polarization of high-order harmonic radiation, *Phys. Rev. A* 76 (3) (2007) 031404(R). <http://dx.doi.org/10.1103/PhysRevA.76.031404>. URL <http://link.aps.org/doi/10.1103/PhysRevA.76.031404>.
- [19] O. Smirnova, Y. Mairesse, S. Patchkovskii, N. Dudovich, D. Villeneuve, P. Corkum, M.Y. Ivanov, High harmonic interferometry of multi-electron dynamics in molecules, *Nature* 460 (7258) (2009) 08253. <http://dx.doi.org/10.1038/nature08253>. URL <http://www.ncbi.nlm.nih.gov/pubmed/19626004>.
- [20] M. Awasthi, A. Saenz, Breakdown of the single-active-electron approximation for one-photon ionization of the  $B^1\Sigma_u^+$  state of  $H_2$  exposed to intense laser fields, *Phys. Rev. A* 81 (6) (2010) 063406. <http://dx.doi.org/10.1103/PhysRevA.81.063406>. URL <http://link.aps.org/doi/10.1103/PhysRevA.81.063406>.
- [21] B. Zhang, Z. Zhao, Strong-field approximation for the ionization of  $N_2$  and  $CO_2$ , *Phys. Rev. A* 82 (3) (2010) 035401. <http://dx.doi.org/10.1103/PhysRevA.82.035401>. URL <http://link.aps.org/doi/10.1103/PhysRevA.82.035401>.
- [22] S. Petretti, Y.V. Vanne, A. Saenz, A. Castro, P. Declava, Alignment-dependent ionization of  $N_2$ ,  $O_2$ , and  $CO_2$  in intense laser fields, *Phys. Rev. Lett.* 104 (22) (2010) 223001. <http://dx.doi.org/10.1103/PhysRevLett.104.223001>. URL <http://link.aps.org/doi/10.1103/PhysRevLett.104.223001>.
- [23] D. Telnov, S.-I. Chu, Effects of multiple electronic shells on strong-field multiphoton ionization and high-order harmonic generation of diatomic molecules with arbitrary orientation: An all-electron time-dependent density-functional approach, *Phys. Rev. A* 80 (4) (2009) 043412. <http://dx.doi.org/10.1103/PhysRevA.80.043412>. URL <http://link.aps.org/doi/10.1103/PhysRevA.80.043412>.
- [24] X. Chu, M. McIntyre, Comparison of the strong-field ionization of  $N_2$  and  $F_2$ : A time-dependent density-functional-theory study, *Phys. Rev. A* 83 (1) (2011) 013409. <http://dx.doi.org/10.1103/PhysRevA.83.013409>. URL <http://link.aps.org/doi/10.1103/PhysRevA.83.013409>.
- [25] M.A. Marques, A. Castro, G.F. Bertsch, A. Rubio, octopus: a first-principles tool for excited electron-ion dynamics, *Comput. Phys. Comm.* 151 (1) (2003) 60–78. [http://dx.doi.org/10.1016/S0010-4655\(02\)00686-0](http://dx.doi.org/10.1016/S0010-4655(02)00686-0).
- [26] U.C.A. Marques, M.A.L. Nogueira, A. Rubio, K. Burke, E.K.U. Gross, Time-Dependent Density Functional Theory, Springer, Heidelberg, Berlin, 2006. <http://dx.doi.org/10.1007/b11767107>.
- [27] D.J. Haxton, K.V. Lawler, C.W. McCurdy, Multiconfiguration time-dependent Hartree–Fock treatment of electronic and nuclear dynamics in diatomic molecules, *Phys. Rev. A* 83 (6) (2011) 063416. <http://dx.doi.org/10.1103/PhysRevA.83.063416>. URL <http://link.aps.org/doi/10.1103/PhysRevA.83.063416>.
- [28] D. Hochstuhl, M. Bonitz, Two-photon ionization of helium studied with the multiconfigurational time-dependent Hartree–Fock method, *J. Chem. Phys.* 134 (8) (2011) 084106. <http://dx.doi.org/10.1063/1.3553176>. URL <http://www.ncbi.nlm.nih.gov/pubmed/21361526>.
- [29] J. Caillat, J. Zanghellini, M. Kitzler, O. Koch, W. Kreuzer, A. Scrinzi, Correlated multielectron systems in strong laser fields: A multiconfiguration time-dependent Hartree–Fock approach, *Phys. Rev. A* 71 (1) (2005) 012712. <http://dx.doi.org/10.1103/PhysRevA.71.012712>. URL <http://link.aps.org/doi/10.1103/PhysRevA.71.012712>.
- [30] K. Kulander, Time-dependent Hartree–Fock theory of multiphoton ionization: Helium, *Phys. Rev. A* 36 (6) (1987) 2726–2738. <http://dx.doi.org/10.1103/PhysRevA.36.2726>. URL <http://link.aps.org/doi/10.1103/PhysRevA.36.2726>.
- [31] B. Zhang, J. Yuan, Z. Zhao, Dynamic core polarization in strong-field ionization of  $CO$  molecules, *Phys. Rev. Lett.* 111 (16) (2013) 163001. <http://dx.doi.org/10.1103/PhysRevLett.111.163001>.
- [32] N. Rohringer, A. Gordon, R. Santra, Configuration-interaction-based time-dependent orbital approach for ab initio treatment of electronic dynamics in a strong optical laser field, *Phys. Rev. A* 74 (4) (2006) 043420. <http://dx.doi.org/10.1103/PhysRevA.74.043420>. URL <http://link.aps.org/doi/10.1103/PhysRevA.74.043420>.
- [33] K. Balzer, S. Bauch, M. Bonitz, Time-dependent second-order Born calculations for model atoms and molecules in strong laser fields, *Phys. Rev. A* 82 (3) (2010) 033427. <http://dx.doi.org/10.1103/PhysRevA.82.033427>. URL <http://link.aps.org/doi/10.1103/PhysRevA.82.033427>.
- [34] M. Abramowitz, I.A. Stegun, Handbook of Mathematical Functions, Dover, 1972.
- [35] T. Rescigno, C. McCurdy, Numerical grid methods for quantum-mechanical scattering problems, *Phys. Rev. A* 62 (3) (2000) 032706. <http://dx.doi.org/10.1103/PhysRevA.62.032706>. URL <http://link.aps.org/doi/10.1103/PhysRevA.62.032706>.
- [36] T.J. Park, J.C. Light, Unitary quantum time evolution by iterative lanczos reduction, *J. Chem. Phys.* 85 (10) (1986) 5870. <http://dx.doi.org/10.1063/1.451548>.
- [37] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J.D. Cruz, A. Greenbaum, S. Hammarling, A. McKenney, D. Sorensen, LAPACK User's Guide, third ed., SIAM, Philadelphia, 1999.
- [38] K.I. Ramchandran, G. Deepa, K. Namboori, Computational Chemistry and Molecular Modeling: Principles and Applications, Springer, Berlin, Heidelberg, 2008.
- [39] L. Tao, C. McCurdy, T. Rescigno, Grid-based methods for diatomic quantum scattering problems: A finite-element discrete-variable representation in prolate spheroidal coordinates, *Phys. Rev. A* 79 (1) (2009) 012719. <http://dx.doi.org/10.1103/PhysRevA.79.012719>. URL <http://link.aps.org/doi/10.1103/PhysRevA.79.012719>.
- [40] B. Zhang, J. Yuan, Z. Zhao, Alignment-dependent ionization of  $H_2^+$ : From multiphoton ionization to tunneling ionization, *Phys. Rev. A* 85 (3) (2012) 033421. <http://dx.doi.org/10.1103/PhysRevA.85.033421>. URL <http://link.aps.org/doi/10.1103/PhysRevA.85.033421>.
- [41] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes, third ed., Cambridge, New York, 2007.
- [42] X. Guan, C. Noble, O. Zatsarinny, K. Bartschat, B. Schneider, Altdse: An Arnoldi-lanczos program to solve the time-dependent Schrödinger equation, *Comput. Phys. Comm.* 180 (12) (2009) 2401–2409. <http://dx.doi.org/10.1016/j.cpc.2009.03.005>.
- [43] M.W. Schmidt, K.K. Baldridge, J.A. Boatz, S.T. Elbert, M.S. Gordon, J.H. Jensen, S. Koseki, N. Matsunaga, K.A. Nguyen, S. Su, T.L. Windus, M. Dupuis, J.A. Montgomery, General atomic and molecular electronic structure system, *J. Comput. Chem.* 14 (11) (1993) 1347–1363. <http://dx.doi.org/10.1002/jcc.540141112>. URL <http://doi.wiley.com/10.1002/jcc.540141112>.
- [44] B. Zhang, Z.X. Zhao, Slimp: Strong laser interaction model package for atoms and molecules, *Comput. Phys. Comm.* <http://dx.doi.org/10.1016/j.cpc.2015.02.031>.
- [45] J. Joachain, N.J. Kylstra, R.M. Potvliege, *Atoms in Intense Laser Fields*, Cambridge, 2012.
- [46] D. Gabor, *J. Inst. Electr. Eng.* 93 (1946) 429.
- [47] B.I. Schneider, J. Segura, A. Gil, X. Guan, K. Bartschat, A new fortran 90 program to compute regular and irregular associated legendre functions, *Comput. Phys. Comm.* 181 (12) (2010) 2091–2097. <http://dx.doi.org/10.1016/j.cpc.2010.08.038>.
- [48] J. Kobus, Finite-difference versus finite-element methods, *Chem. Phys. Lett.* 202 (1) (1993) 7.
- [49] Nist cccbdb website, <http://cccbdb.nist.gov/expdata.asp>.
- [50] A. Staudte, S. Patchkovskii, D. Pavičić, H. Akagi, O. Smirnova, D. Zeidler, M. Meckel, D. Villeneuve, R. Dörner, M. Ivanov, P. Corkum, Angular tunneling ionization probability of fixed-in-space  $H_2$  molecules in intense laser pulses, *Phys. Rev. Lett.* 102 (3) (2009) 1–4. <http://dx.doi.org/10.1103/PhysRevLett.102.033004>. URL <http://link.aps.org/doi/10.1103/PhysRevLett.102.033004>.
- [51] X.M. Tong, Z.X. Zhao, C.D. Lin, Correlation dynamics between electrons and ions in the fragmentation of  $D_2$  molecules by short laser pulses, *Phys. Rev. A* 68 (4) (2003) 043412. <http://dx.doi.org/10.1103/PhysRevA.68.043412>. URL <http://link.aps.org/doi/10.1103/PhysRevA.68.043412>.
- [52] C.D. Lin, X.M. Tong, Z.X. Zhao, Effects of orbital symmetries on the ionization rates of aligned molecules by short intense laser pulses, *J. Modern Opt.* 53 (1–2) (2006) 21–33. <http://dx.doi.org/10.1080/09500340500159492>.
- [53] B. Zhang, J. Yuan, Z. Zhao, Dynamic orbitals in high-order harmonic generation from  $CO$  molecules, *Phys. Rev. A* 90 (3) (2014) 035402. <http://dx.doi.org/10.1103/PhysRevA.90.035402>.