ezDyson User's Manual

Version 4.0, July 2017

Samer Gozem and Anna I. Krylov

Department of Chemistry, University of Southern California, Los Angeles, California 90089-0482

The *ezDyson* program is available for download at: `iopenshell.usc.edu/downloads/` Please contact Prof. Anna I. Krylov (krylov@usc.edu) if you have questions, problems, or bugs to report.

Contents

I. ABOUT *EZDYSON*

ezDyson is a C++ code that calculates absolute photodetachment/photoionization cross sections, photoelectron angular distributions (PADs), and anisotropy parameters (β) using Dyson orbitals computed by an *ab initio* program. The calculation is based on the following approximations: (1) weak field limit, i.e., the photodetachment or photoionization is treated in a perturbative regime; (2) dipole approximation, i.e., assuming that the wavelength of the radiation field is longer than the size of the molecule and that the process is one-photon (first order in perturbation theory); (3) sudden approximation, i.e., we ignore the interactions between the ejected photoelectron and the remaining core electrons (except the electrostatic interaction, which can be accounted for by using Coulomb waves); (4) strong orthogonality condition, i.e., the continuum orbital is orthogonal to all states of initial system, $\Psi(N)$. We use either plane waves or Coulomb waves to describe the continuum state of the photoelectron.

The code requires information from a quantum chemical calculation (geometries, atomic basis set), details of the experiment (laser polarization, ionization energy, molecular orientation, range of electron kinetic energies), the coefficients of the Dyson orbitals in the atomic orbital (AO) basis, and the norm of the Dyson orbitals. The *ezDyson* input is in *xml* format; it can be prepared manually. Alternatively, a *Python* interface can be used to extract data from an electronic structure program and prepare an xml input automatically. A *Python* interface is currently provided only for *Q-Chem* outputs. Dyson orbital calculations are implemented within the equation-of-motion coupled-cluster (EOM-CC) suite of methods[1] in *Q-Chem*[2].

This manual provides a detailed description of the input structure and gives a number of examples. It also presents a derivation of the working expressions using a time-independent treatment of the photodetachment/photoionization based on Dyson orbitals and angular momentum theory. The structure of the text is as follows: Section II presents a brief explanation of the model and working expressions, Section III explains how to install and run the program, Section IV explains the input structure and the relevant keywords, Section V describes the output structure.

A.   Caveats

- Our early implementation of the Dyson orbitals (in the *ccman* module of *Q-Chem*) had a bug leading to incorrect norms. A correct implementation is in the *ccman2* module of *Q-Chem* which was released in *Q-Chem* version 4.3. Please use the latest version of Q-Chem for quantavely accurate photoionization/photodetachment cross sections.

- *Python* scripts provided in this *ezDyson* distribution are only compatible with with the *ccman2* Dyson code which is available in *Q-Chem* version 4.3 or later. Consult *Q-Chem's* user manual for EOM-CC and Dyson orbital calculations[3].

B.   What's new in *ezDyson 3 and 4*

A number of bottlenecks and important bugs in an early *ezDyson* code have been corrected. The most important differences include new procedures for isotropic averaging, the ability of the new code to correcly compute cross sections with Coulomb waves, and an implementation that avoids using the plane wave expansion as required for certain applications. Specific changes in the code include:

- The expression for total and differential cross sections in `dyson_main.C` and `wavetypes.C` are now consistent with the derivation in Section II of the manual.

- Fixed bugs related to reading of Dyson orbitals in the AO basis, which was leading to incorrect norms.

- Added the ability to read basis sets with g-functions (pure and Cartesian angular momentum).

- Fixed compilation error related to undeclared "theSPH".

- Several improvements to the efficiency of the code.

- Fixed a bug related to computing cross sections with Coulomb waves.

- Version 3.3 of *ezDyson* introduces a new analytical averaging scheme for total and differential cross sections, described in Section II.

- Version 4.0 introduces a new numerical averaging approach that circumvents the need to use the plane wave expansion (see Section II F for details).

- A number of changes to the input and output structure have been made. See Sections IV and V for details.

C. Acknowledgments

II. THEORETICAL BACKGROUND

A. Absolute cross sections from Dyson orbitals

We use atomic units in all derivations and in the code. The expression for calculating the differential photodetachment/photoionization cross section, $\frac{d\sigma}{d\Omega_k}$, is [4, 5]:

$$\frac{d\sigma}{d\Omega_k} = \frac{4\pi^2 kE}{c} \left| D_k^{IF}(\theta, \phi) \right|^2,\tag{1}$$

where k is the magnitude of the photoelectron wave vector $\boldsymbol{k}$, E is the energy of the ionizing radiation, and c is the speed of light. D_k^{IF} is the photoelectron dipole matrix element:

$$D_k^{IF} = \boldsymbol{u} \langle \Psi_I^N | \boldsymbol{r} | \Psi_F^{N-1} \Psi_k^{el} \rangle,\tag{2}$$

where $\boldsymbol{r}$ is the dipole moment operator and $\boldsymbol{u}$ is a unit vector in the direction of the polarization of light. Assuming strong orthogonality conditions, Eq. (2) becomes:

$$D_k^{IF} = \boldsymbol{u} \langle \phi^d | \boldsymbol{r} | \Psi_k^{el} \rangle,\tag{3}$$

where Ψ_k^{el} is the photoelectron wave function and ϕ^d is a Dyson orbital connecting the initial (N-electron) and final (N-1-electron) states:

$$\phi^d(1) = \sqrt{N} \int \Psi_I^N(1,\ldots,n)\Psi_F^{N-1}(2,\ldots,n)d2\ldots dn. \tag{4}$$

Eq. (3) can be expressed as a sum of its x, y and z components:

$$D_k^{IF} = \sum_{\alpha=x,y,z} \langle \phi^d|r_\alpha|\Psi_k^{el}\rangle cos(\boldsymbol{u}, e_\alpha), \tag{5}$$

Here, r_α denotes the x, y, or z component of the dipole moment operator, and the *cos* term represents the respective projections of the light polarization vector.

In the non-Hermitian EOM-CC theory, the expression for $|D_k^{IF}|^2$ depends on left and right Dyson orbitals [6] and is given by:

$$|D_k^{IF}|^2 = \sum_{\alpha=x,y,z} \langle \phi_L^d|r_\alpha|\Psi_k^{el}\rangle\langle\Psi_k^{el}|r_\alpha|\phi_R^d\rangle cos^2(\boldsymbol{u}, e_\alpha) = \sum_{\alpha=x,y,z} |D_k^{IF}|_\alpha^2 cos^2(\boldsymbol{u}, e_\alpha). \tag{6}$$

The photoelectron wave function, Ψ_k^{el}, can be treated as a plane wave:

$$\Psi_k^{el} = \frac{1}{(2\pi)^{3/2}}e^{i\boldsymbol{k}\cdot\boldsymbol{r}}. \tag{7}$$

The $(2\pi)^{-3/2}$ factor is the "continuum normalization" typically used for plane waves [7]. Plane waves can be written as a sum of spherical waves (this is known as the plane-wave expansion) [8]:

$$e^{i\boldsymbol{k}\cdot\boldsymbol{r}} = 4\pi \sum_{l=0}^{\infty} \sum_{m=-l}^{l} i^l R_l(kr)Y_{lm}(\boldsymbol{r})Y_{lm}(\boldsymbol{k}). \tag{8}$$

Each spherical wave is characterized by its energy, $E = \frac{k^2}{2m}$, and angular momentum, l, m, and is a product of a radial function $R_l(kr)$ and spherical spherical harmonic functions Y_{lm}. $Y_{lm}(\boldsymbol{r})$ depends on the position vector $\boldsymbol{r}$ and therefore determines the shape of the plane wave in position space, while $Y_{lm}(\hat{k})$ determines the direction in which the plane wave travels.

We use orthonormal spherical harmonics:

$$Y_{lm}(\theta, \phi) = \Theta_{lm}(\theta)\Phi_m(\phi), \tag{9}$$

$$\int_0^{2\pi} \Phi_m^*(\phi)\Phi_m(\phi)d\phi = \delta_{mm'}, \tag{10}$$

$$\int_0^\pi Y_{l'm'}(\theta, \phi)Y_{lm}(\theta, \phi)sin\theta d\theta d\phi = \delta_{ll'}\delta_{mm'}. \tag{11}$$

For a plane wave, the radial function is a spherical Bessel function ($R_l(kr) = j_l(kr)$). Using a plane wave is appropriate when there is electrostatic interaction beteen the ejected photoelectron and the remaining core. However, when the ionized core is charged and the electrostatic interaction between the photoelectron and core cannot be ignored, one can use a Coulomb wave instead of a plane wave. The Coulomb wave can also be expanded into a sum of Coulomb partial waves using the same expression as Eq. (8), but instead of using spherical Bessel functions, the radial part is given by a Coulomb radial wave function[9]:

$$R_l(kr, \eta) = (2kr)^l e^{-\frac{\pi\eta}{2}} \frac{|\Gamma(l+1+i\eta)|}{\Gamma(2l+2)} e^{-ikr} \, _1F_1(l+1-i\eta, 2l+2, 2ikr), \qquad (12)$$

where Γ is the Gamma function and $_1F_1$ is the confluent hypergeometric function of the first kind. η is the Sommerfeld parameter, which is equal to $-Z/k$ in atomic units, where Z is the charge of the ionized core and k is the magnitude of the photoelectron wave vector. When $Z=0$, the Coulomb wave becomes a plane wave since Eq. (12) reduces to a spherical Bessel function.

From Eqs. (6), (7) and (8) we obtain (see Ref. [10]):

$$\begin{aligned} \left|D_k^{IF}\right|_\alpha^2 &= \frac{(4\pi)^2}{(2\pi)^3} \sum_{l,l'=0}^{\infty} \sum_{m=-l}^{l} \sum_{m'=-l'}^{l'} \langle \phi_L^d | r_\alpha | i^l R_l(kr) Y_{lm}(\hat{r}) \rangle \\ &\times \langle i^{l'} R_{l'}(kr) Y_{l'm'}(\hat{r}) | r_\alpha | \phi_R^d \rangle Y_{lm}(\hat{k}) Y_{l'm'}^*(\hat{k}). \end{aligned} \qquad (13)$$

We define $C_{klm}^L[\alpha]$ and $C_{kl'm'}^R[\alpha]$ as:

$$\begin{aligned} C_{klm}^L[\alpha] &= i^l \int \phi_L^d(\mathbf{r}) r_\alpha R_l(kr) Y_{lm}(\theta, \phi) d\mathbf{r}, \\ C_{kl'm'}^R[\alpha] &= i^{l'} \int \phi_R^d(\mathbf{r}) r_\alpha R_{l'}(kr) Y_{l'm'}(\theta, \phi) d\mathbf{r}. \end{aligned} \qquad (14)$$

The C_{klm}s give the amplitude of each spherical wave that contributes to the final wave function of the photoelectron. From Eqs. (13) and (14):

$$\left|D_k^{IF}\right|_\alpha^2 = \frac{2}{\pi} \sum_{l,l'=0}^{\infty} \sum_{m=-l}^{l} \sum_{m'=-l'}^{l'} C_{klm}^L[\alpha] \, C_{kl'm'}^R[\alpha] \, Y_{lm}(\hat{k}) Y_{l'm'}^*(\hat{k}). \qquad (15)$$

Eq. (15) provides the photoelectron dipole matrix elements between the Dyson orbital and the photoelectron wave function, which allows us to ultimately calculate the total cross sections, differential cross sections (for PADs), and β anisotropy parameter. Eq. (14) is evaluated by numerical integration on a grid using Dyson orbitals computed by a quantum chemistry program (given as an expansion over the atom-centered Gaussian basis functions).

From Eqs. (1) and (15), we obtain the expression for the differential cross section. At this point we also multiply by the Dyson orbital norms (not shown in the equation below):

$$\frac{d\sigma}{d\Omega_k}|_\alpha = \frac{8\pi kE}{c} \sum_{l,l'=0}^{\infty} \sum_{m=-l}^{l} \sum_{m'=-l'}^{l'} C_{klm}^{L}[\alpha]\, C_{kl'm'}^{R}[\alpha]\, Y_{lm}(\hat{k})Y_{l'm'}^{*}(\hat{k}). \tag{16}$$

At this point we also multiply by the Dyson orbital norms and by degneracy factors (to account for spin and orbital degeneracy, see Section II C), although these factors are not shown in Eq. (16) above.

Integration over all $d\Omega$ provides the total photodetachment/photoionization cross section (see Eq. (11)):

$$\sigma[\alpha] = \frac{8\pi kE}{c} \sum_{klm} |C_{klm}[\alpha]|^2. \tag{17}$$

The derivation above does not include averaging over molecular orientations, and therefore is appropriate for a molecule fixed in a specific position in the lab frame and with a specific light polarization that has components e_x, e_y, and e_z. The differential cross section and total cross section in this case is simply the sum of the three x, y, and z components of $\frac{d\sigma}{d\Omega_k}$ and σ, respectively. However, in the case of an isotropic sample that consists of randomly oriented molecules, one has to average over all molecular orientations. The way to do this is explained in the following section.

B. Averaging over molecular orientations

The three components of $\sigma[\alpha]$ describe contributions to the total cross section from the light polarized along e_α from a molecule at a fixed orientation. In order to average over all molecular orientations, one can instead average over all possible orientations of light polarization, $\boldsymbol{u}$. By doing so, one finds that the total cross section for isotropic averaging is simply the average of the x, y, and z components of the cross section. This is exactly the result derived in Ref. [11]. Note that for a single-photon experiment, there is no difference in the total cross section computed using polarized versus non-polarized light [11]. Thus, no additional factors should appear due to using polarized light, contrary to some remarks in the literature.

This simple averaging procedure works for total cross sections, but it is not appropriate for differential cross sections, PADs, and beta anisotropy parameters. Here we present a

more general averaging procedure that works for differential cross sections as well as total cross sections. This is the averaging procedure implemented in *ezDyson*.

We begin with Eq. (13) and recognize that some quantities belong to the molecular frame and others to the lab frame. In particular, the direction of the polarization of light ($\mathbf{r}$) is in the lab frame. Also, $Y_{lm}(\boldsymbol{k})$ and $Y_{l'm'}^*(\boldsymbol{k})$ in Eq. (15), which indicate the direction that the photoelectron travels, must be obtained in the lab frame (because the electrons are detected in the lab frame). All other quantities are in the molecular frame. We will, for the most part, follow the derivation presented in Ref. [12] for the averaging scheme.

To rotate spherical harmonics from one frame to another we make use of Wigner D-matrices, which give the matrix elements of the rotation operator R [13]:

$$\langle l, m \,|\mathcal{R}(\alpha, \beta, \gamma)|\, l', m'\rangle = \delta_{ll'} D_{m,m'}^l(\alpha, \beta, \gamma), \tag{18}$$

where $\mathcal{R}(\alpha, \beta, \gamma)$ is a 3D rotation operator using Euler's z-y-z ("yaw-pitch-roll") convention to define α, β, and γ. Applying the 3D rotation operator to a spherical harmonic function $|l, m'\rangle$ to rotate it to a new state $|\overline{l, m'}\rangle$ and using resolution of the identity ($I = \sum_m |l, m\rangle\langle l, m|$) gives:

$$\begin{aligned} |\overline{l, m'}\rangle &= \mathcal{R}(\alpha, \beta, \gamma)|l, m'\rangle, \\ |\overline{l, m'}\rangle &= \sum_m |l, m\rangle\langle l, m|\mathcal{R}(\alpha, \beta, \gamma)|l, m'\rangle, \\ |\overline{l, m'}\rangle &= \sum_m D_{m,m'}^l |l, m\rangle. \end{aligned} \tag{19}$$

Eq. (19) allows us to express spherical harmonics in the molecular frame as linear combinations of spherical harmonics in the laboratory frame. [14]:

$$Y_{lm'}(\theta', \phi') = \sum_{m=-l}^{l} D_{m,m'}^l(\alpha, \beta, \gamma) Y_{lm}(\theta, \phi). \tag{20}$$

Note that in Eq. (20) we use θ' and ϕ' to represent functions in the molecular frame, while θ and ϕ represent functions in the lab frame. α, β, and γ are Euler's angles that relate the two frames to each other. We will use this convention for the rest of this section. We can also express spherical harmonics in the lab frame as a linear combination of spherical harmonics in the molecular frame by using the inverse of Eq. (20):

$$Y_{lm}(\theta, \phi) = \sum_{m'=-l}^{l} D_{m,m'}^l(\alpha, \beta, \gamma) Y_{lm'}(\theta', \phi'). \tag{21}$$

Below, we make use of two properties of Wigner D-matrices. First, a product of two Wigner D-matrices can be expressed a Clebsch-Gordan series [15]:

$$D^{l_1}_{m_1 k_1}(\alpha, \beta, \gamma) D^{l_2}_{m_2 k_2}(\alpha, \beta, \gamma) = \sum_{L=|l_1-l_2|}^{l_1+l_2} \sum_{M=-L}^{L} \sum_{K=-L}^{L} C^{l_1 \, l_2 \, L}_{m_1 \, m_2 \, M} \, C^{l_1 \, l_2 \, L}_{k_1 \, k_2 \, K} \, D^L_{MK}(\alpha, \beta, \gamma), \quad (22)$$

where $C^{l_1 \, l_2 \, L}_{m_1 \, m_2 \, M}$ (and $C^{l_1 \, l_2 \, L}_{k_1 \, k_2 \, K}$) are the Clebsch-Gordan coefficients. These are numbers (real numbers if using the Condon Shortley phase convention) that arise in angular momentum coupling. The Clebsch-Gordan coefficients are equal to zero unless $|l_1 - l_2| \leq L \leq l_1 + l_2$, and $M = m_1 + m_2$.

The second property is the orthogonality relations Wigner D-matrices:

$$\int_0^{2\pi} d\alpha \int_0^{\pi} d\beta \int_0^{2\pi} d\gamma \; D^{l_1*}_{m_1 k_1}(\alpha, \beta, \gamma) D^{l_2}_{m_2 k_2}(\alpha, \beta, \gamma) = \frac{8\pi^2}{2l_1 + 1} \delta_{m_1 m_2} \delta_{k_1 k_2} \delta_{l_1 l_2}. \quad (23)$$

We now revisit Eq. (13), which was written for a specific polarization vector r_α. Intead of r_α, we will use light polarized along the z axis of the lab frame (we used the z axis for simplicity, the axis we choose does not matter). We can write z dipole moment operator in terms of spherical harmonics:

$$z = \left(\frac{4\pi}{3}\right)^{\frac{1}{2}} r Y_{10}(\theta, \phi). \quad (24)$$

Eq. (24) simply follows from rearranging the definition of the Y_{10} spherical harmonic function. Using Eq. (21), we can rewrite Eq. (24) in the molecular frame:

$$z = \left(\frac{4\pi}{3}\right)^{\frac{1}{2}} r \sum_{m_2=-1}^{1} D^1_{0m_2}(\alpha, \beta, \gamma) Y_{1m_2}(\theta', \phi'). \quad (25)$$

By expressing light polarization in the molecular frame, everything becomes consistently in the molecular frame. However, as mentioned above, we need the final D^{IF}_k to be in the lab frame. Therefore, we need to express $Y_{lm}(\hat{k})$ in the lab frame using the relation in Eq. (20):

$$Y_{lm}(\theta'_k, \phi'_k) = \sum_{m_1=-l}^{l} D^l_{m_1, m}(\alpha, \beta, \gamma) Y_{lm_1}(\theta_k, \phi_k). \quad (26)$$

Using Eq. (13) but replacing r_α by Eq. (24) and using Eq. (26), we obtain:

$$|D^{IF}_k(\alpha, \beta, \gamma, \theta_k, \phi_k)|^2 = \frac{2}{\pi} \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \sum_{m_1=-l}^{l} \sum_{m_2=-1}^{1} \left(\left(\frac{4\pi}{3}\right)^{\frac{1}{2}} i^l D^{1*}_{0m_2}(\alpha, \beta, \gamma) D^{l*}_{m_1, m}(\alpha, \beta, \gamma) \right.$$

$$\left. \int \phi^d_L(\mathbf{r}) r R_l(kr) Y^*_{1m_2}(\theta', \phi') Y^*_{lm}(\theta', \phi') d\mathbf{r} \right) Y_{lm_1}(\theta_k, \phi_k)$$

$$\times \sum_{l'=0}^{\infty} \sum_{m'=-l'}^{l'} \sum_{m_1'=-l'}^{l'} \sum_{m_2'=-1}^{1} \left(\left(\frac{4\pi}{3}\right)^{\frac{1}{2}} i^{l'} D_{0m_2'}^{1}(\alpha,\beta,\gamma) D_{m_1',m'}^{l'}(\alpha,\beta,\gamma) \right.$$

$$\left. \int \phi_R^d(\mathbf{r}) r R_{l'}(kr) Y_{1m_2'}(\theta',\phi') Y_{l'm'}(\theta',\phi') d\mathbf{r} \right) Y_{l'm_1'}^*(\theta_k,\phi_k)$$

$$= \frac{8}{3} \sum_{lmm_1m_2} \sum_{l'm'm_1'm_2'} C_{klmm_2}^{L} C_{kl'm'm_2'}^{R} D_{0m_2}^{1*}(\alpha,\beta,\gamma) D_{m_1,m}^{l*}(\alpha,\beta,\gamma)$$

$$D_{0m_2'}^{1}(\alpha,\beta,\gamma) D_{m_1',m'}^{l'}(\alpha,\beta,\gamma) Y_{lm_1} Y_{l'm_1'}^*, \tag{27}$$

where we have defined $C_{klmm_2}^L$ and $C_{kl'm'm_2'}^R$ as:

$$C_{klmm_2}^{L} = i^l \int \phi_L^d(\mathbf{r}) r R_l(kr) Y_{1m_2}^*(\theta',\phi') Y_{lm}^*(\theta',\phi') d\mathbf{r},$$

$$C_{klmm_2}^{L} = i^l \int \phi_R^d(\mathbf{r}) r R_l(kr) Y_{1m_2}(\theta',\phi') Y_{lm}(\theta',\phi') d\mathbf{r}. \tag{28}$$

$C_{klmm_2}^L$ and $C_{kl'm'm_2'}^R$ are not to be confused with Clebsch-Gordan coefficients, which instead have 3 superscripts and 3 subscripts. Using the Wigner D-matrix coupling equation (Eq. (22)) to couple the D^l and D^1 pair of terms we obtain:

$$|D_k^{IF}(\alpha,\beta,\gamma,\theta_k,\phi_k)|^2 = \frac{8}{3} \sum_{lmm_1m_2} \sum_{l'm'm_1'm_2'} C_{klmm_2}^{L} C_{kl'm'm_2'}^{R}$$

$$\times \sum_{L=|l-1|}^{l+1} C_{m\ m_2\ m+m_2}^{l\ 1\ L} C_{m_1\ 0\ m_1}^{l\ 1\ L} D_{m+m_2,m_1}^{L*}(\alpha,\beta,\gamma)$$

$$\times \sum_{L'=|l'-1|}^{l'+1} C_{m'\ m_2'\ m'+m_2'}^{l'\ 1\ L'} C_{m_1'\ 0\ m_1'}^{l'\ 1\ L'} D_{m'+m_2',m_1'}^{L'}(\alpha,\beta,\gamma)$$

$$\times Y_{lm_1}(\theta_k,\phi_k) Y_{l'm_1'}^*(\theta_k,\phi_k). \tag{29}$$

Finally, to average over all molecular orientations, we integrate Eq. (29) over all Euler angles (α, β, and γ) and use the orthogonality relations of Wigner D-matrices (Eq. (23)) to arrive to:

$$\overline{|D_k^{IF}(\theta_k,\phi_k)|^2} = \frac{8}{3} \sum_{lmm_1m_2} \sum_{l'm'm_1'm_2'} (C_{klmm_2}^{L} C_{kl'm'm_2'}^{R} Y_{lm_1}(\theta_k,\phi_k) Y_{l'm_1'}^*(\theta_k,\phi_k)$$

$$\times \sum_{L=|l-1|}^{l+1} \sum_{L'=|l'-1|}^{l'+1} C_{m\ m_2\ m+m_2}^{l\ 1\ L} C_{m_1\ 0\ m_1}^{l\ 1\ L} C_{m'\ m_2'\ m'+m_2'}^{l'\ 1\ L'} C_{m_1'\ 0\ m_1'}^{l'\ 1\ L'}$$

$$\times \frac{8\pi^2}{2L+1} \delta_{LL'} \delta_{m_1m_1'} \delta_{m+m_2,m'+m_2'}. \tag{30}$$

Plugging Eq. (30) into Eq. (1) gives the final expression for the differential cross section. In *ezDyson*, the C_{klmm_2} coefficients defined in Eq. (28) are computed by numerical integration on a grid. The Clebsch-Gordan coefficients have been pre-computed up to $l_{max} = 10$ using the algorithms from Refs. [16–18] and then simply looked up when needed within ezDyson.

C. Accounting for orbital and spin degeneracies

To compute the total cross section for a system, one needs to account for multiple target states. If several target states F are accessible, (i.e., multiple ionized states when using sufficiently high energy or when the target has multiple degenerate states), then one must account for their contributions to the cross section:

$$\frac{d\sigma}{d\Omega_k} = \frac{4\pi^2 E}{c} \cdot \sum_F k_F \left| D_k^{IF}(\theta, \phi) \right|^2. \tag{31}$$

Different target states give rise to different Dyson orbitals, according to Eq. (4). For non-degenerate target states, one simply needs to compute cross sections for the individual states separately and add them up. Spherically averaged cross sections for systems with symmetry related degeneracies can be computed by calculating the cross section corresponding to one degenerate state and multiplying it by the number of the states in the manifold. That is, the cross section for photodetachment from F$^-$ corresponding to the F(^{2}P) channel is:

$$\sigma(^2P) = 3 \times \sigma(\phi^d \equiv p_z) \tag{32}$$

Likewise, for ionizations from π orbitals in CO or acetylene, the cross sections should be multiplied by 2.

One needs also to account for spin-degeneracy of the target states. For ionization of closed-shell singlet states, the cross section should be multiplied by 2. The derivation is given below.

For ionization of a singlet state, the target state of the ionized system is a singlet-coupled $\Psi_{N-1}^F \Psi_k^{el}$ giving rise to the following expression for $|D_k^{IF}|$:

$$
\begin{aligned}
|D_k^{IF}|^2 &= \left| \langle \Psi_N^I \, |\mu| \, \frac{1}{\sqrt{2}} \left(\Psi_{N-1}^F(\alpha)\Psi_k^{el}(\beta) - \Psi_{N-1}^F(\beta)\Psi_k^{el}(\alpha) \right) \rangle \right|^2 \\
&= \frac{1}{2} \Big(\langle \Psi_N^I(\alpha) \, |\mu| \, \Psi_{N-1}^F(\beta)\Psi_k^{el}(\alpha) \rangle^2 + \langle \Psi_N^I(\beta) \, |\mu| \, \Psi_{N-1}^F(\alpha)\Psi_k^{el}(\beta) \rangle^2 \\
&\quad -2\langle \Psi_N^I(\alpha) \, |\mu| \, \Psi_{N-1}^F(\beta)\Psi_k^{el}(\alpha) \rangle \langle \Psi_N^I(\beta) \, |\mu| \, \Psi_{N-1}^F(\alpha)\Psi_k^{el}(\beta) \rangle \Big) \\
&= \frac{1}{2} \left(\langle \phi^d(\beta) \, |\mu| \, \Psi_k^{el}(\beta) \rangle^2 + \langle \phi^d(\alpha) \, |\mu| \, \Psi_k^{el}(\alpha) \rangle^2 - 2\langle \phi^d(\beta) \, |\mu| \, \Psi_k^{el}(\beta) \rangle \langle \phi^d(\alpha) \, |\mu| \, \Psi_k^{el}(\alpha) \rangle \right) \\
&= 2 \left| \langle \phi^d(x) \, |\mu| \, \Psi_k^{el}(x) \rangle \right|^2,
\end{aligned}
$$

where $\phi^d(x)$ is a spatial part of the Dyson orbital defined by Eq. (4). Note that α and β Dyson orbitals have different signs because a pair α and β electrons need to be swapped either in Ψ^I or Ψ^F when applying Eq. (4).

D. Accounting for Franck-Condon factors (FCFs)

In molecules, it may be important to account for the vibrational wave functions of the initial and final states by computing the photoelectron dipole matrix element, D_k^{IF}, for each vibronic transition and then using these in Eq. (1). Within the Born-Oppenheimer approximation, we obtain:

$$D_{nn'k}^{IF} = \boldsymbol{u} \langle \Psi_I^N \xi_n^I | \boldsymbol{r} | \Psi_F^{N-1} \xi_{n'}^F \Psi_k^{el} \rangle, \tag{33}$$

where ξ_n^I and $\xi_{n'}^F$ are the vibrational wave functions of the nth vibrational level of the initial electronic state and the $n'^{\,th}$ vibrational level of the final electronic state, respectively. Within the Condon approximation (i.e., assuming that the electronic wave functions depend weakly on nuclear coordinates so that the photoelectron dipole matrix element can be evaluated at a fixed nuclear geometry, R_0, corresponding to the equilibrium geometry of the initial state), we obtain:

$$
\begin{aligned}
D_{nn'k}^{IF} &= \boldsymbol{u} \langle \Psi_I^N(R_0) \xi_n^I(R) | \boldsymbol{r} | \Psi_F^{N-1}(R_0) \xi_{n'}^F(R) \Psi_k^{el} \rangle, \\
D_{nn'k}^{IF} &= \boldsymbol{u} \langle \Psi_I^N(R_0) | \boldsymbol{r} | \Psi_F^{N-1}(R_0) \Psi_k^{el} \rangle \langle \xi_n^I(R) | \xi_{n'}^F(R) \rangle, \\
D_{nn'k}^{IF} &= D_k^{IF}(R_0) \langle \xi_n^I(R) | \xi_{n'}^F(R) \rangle.
\end{aligned}
\tag{34}
$$

The square of the photoelectron dipole matrix element is therefore:

$$\left| D_{nn'k}^{IF} \right|^2 = \left| D_k^{IF} \right|^2 \langle \xi_n^I | \xi_{n'}^F \rangle^2. \tag{35}$$

The $\langle \xi_n^I | \xi_{n'}^F \rangle$ term is called the Franck-Condon overlap integral factor (FCF), and can be computed by an independent program, *ezSpectrum* [23].

By combining Eqns. (1) and (35), we obtain:

$$\frac{d\sigma}{d\Omega_k} = \frac{4\pi^2}{c} E \sum_{nn'} P_n \left| D_{nn'k}^{IF} \right|^2 = \frac{4\pi^2}{c} E \sum_{nn'} P_n \left| D_k^{IF} \right|^2 \langle \xi_n^I | \xi_{n'}^F \rangle^2, \tag{36}$$

where P_n, the population of vibrational level n, was added to account for the potential thermal population of initial vibrational states. The sum of all P_n is 1. E in Eq. (36) is the energy of the ionizing radiation, which is equal to the energy of the vibronic transition plus the kinetic energy of the photoelectron (E_k). Thus, the latter is:

$$E_k = \frac{k_{nn'}^2}{2} = E - IE_{00}^{IF} - (E_{vib}^F(n) - E_{vib}^I(n')) = E - IE_{00}^{IF} - \Delta E_{vib}(n, n'), \tag{37}$$

where E is, as before, the energy of the ionizing radiation, IE_{00}^{IF} is the ZPE corrected adiabatic ionization/detachment energy of the initial state, and $E_{vib}^i(n)$ and $E_{vib}^f(n')$ are the excess vibrational energies of the initial and final states above $n = 0$ and $n' = 0$, respectively.

Eq. (36) requires summing over D_k^{IF} for different vibronic transitions. Since D_k^{IF} is only computed for discrete values of E_k, performing the simulation is not entirely trivial, so we have provided a script that computes the result of Eq. (36) for discrete energies automatically. Details on how to use this script are in Section V C.

E.   Computing PADs and β anisotropy

Information about the PAD can be obtained by looking at the differential cross section obtained from Eq. (16). For PAD calculations in the molecular frame, the laser polarization must be along the z-axis, since then one only needs to consider the dependence of the cross section on θ and not ϕ. $ezDyson$ computes the cross section at 101 values of θ at equal intervals and prints them in a separate output file. These can be plotted to view the angular distribution of the photoelectrons as a function of θ. The β anisotropy parameter can also be computed by looking at the value of the differential cross section at $\theta = 0$ (σ_{par}) and $\theta = \frac{\pi}{2}$ (σ_{perp}). The expression used to compute the β anisotropy parameter is [10]:

$$\beta = \frac{2(\sigma_{par} - \sigma_{perp})}{\sigma_{par} + 2\sigma_{perp}}. \tag{38}$$

If the calculation is in the lab frame, the averaging scheme in Section II B is used to compute spherically averaged PADs and anisotropy parameters. For molecular frame calculations, on the other hand, (σ_{par}) and (σ_{perp}) are computed for a specific orientation of the molecule assuming that z-polarized light is used. Therefore, for non-averaged PAD and β calculations, the user must specify z-polarized light in the $ezDyson$ input.

F.   Total cross sections and β without the plane wave expansion

Using the plane wave expansion of Eq. (8) in the calculation of the photoelectron dipole matrix element is useful for a number of reasons. It provides an important tool to analyze the wave function of the ejected photoelectron in terms of angular momentum quantum numbers.

The plane wave expansion also allows the derivation of an exact analytical expressions for averaging over molecular orientations, as demonstrated in Section II B.

On the other hand, the plane wave expansion is an infinite expansion over all angular momentum quantum numbers l. In practice, we truncate the expansion at some value, l_{max}. In atomic systems, where selection rules apply, such a truncation is justified (see "input file structure" section below for an explanation). Even in molecules, such a truncation may be justified by for Dyson orbitals that resemble atomic orbitals, or by the fact that large angular momentum contributions are typically small at low photoelectron kinetic energies. However, at high energies, the plane wave expansion approach becomes problematic as contributions from large angular momentum are no longer negligible. For high energy regimes, therefore it is desirable to have expressions for cross sections and β anisotropies that do not rely on the plane wave expansion.

From Eqs. (3) and (7), we obtain:

$$|D_k^{IF}|^2_{par} = \frac{1}{(2\pi)^3} \langle \phi_L^d | \boldsymbol{r}_u | e^{i\boldsymbol{z}\cdot\boldsymbol{r}} \rangle \langle e^{-i\boldsymbol{z}\cdot\boldsymbol{r}} | \boldsymbol{r}_u | \phi_R^d \rangle, \tag{39}$$

$$|D_k^{IF}|^2_{perp} = \frac{1}{(2\pi)^3} \left(\frac{1}{2} \langle \phi_L^d | \boldsymbol{r}_u | e^{i\boldsymbol{x}\cdot\boldsymbol{r}} \rangle \langle e^{-i\boldsymbol{x}\cdot\boldsymbol{r}} | \boldsymbol{r}_u | \phi_R^d \rangle + \frac{1}{2} \langle \phi_L^d | \boldsymbol{r}_u | e^{i\boldsymbol{y}\cdot\boldsymbol{r}} \rangle \langle e^{-i\boldsymbol{y}\cdot\boldsymbol{r}} | \boldsymbol{r}_u | \phi_R^d \rangle \right). \tag{40}$$

Eq. (39) and (40) give parallel and perpendicular cross sections, which can be used to get the β anisotropies using Eq. (38) and the total cross sections using the following expression:

$$\sigma = \frac{4\pi}{3} \left(\sigma_{par} + 2\sigma_{perp} \right). \tag{41}$$

The averaging is done numerically. Specifically, σ_{par} and σ_{perp} are calculated at orientations corresponding to 150 equally distributed points on the surface of a sphere precomputed with the program REPULSION [19].

III.   RUNNING *EZDYSON*

A.   Installation

To extract the files, execute:

```
tar -xzf ezdyson.v3.tar.gz
```

A general Linux compiled executable is included (`exedys`). Makefiles for different platforms are also provided in the main directory. If necessary to recompile for a different platform, copy the `mk` file for the proper platform into `makefile` and run:

```
make clean
```

then:

```
make
```

B.   Using *ezDyson* with the GNU Scientific Library

ezDyson has a built-in code for running calculations with the photoelectron wave function treated as a Coulomb wave. This code is adapted from the PL/1 code in Ref. [20]. However, this code may be CPU intensive (especially for smaller kinetic energies and higher angular momenta). It can also lead to "N/A" or incorrect cross sections if very small kinetic energies are used. To prevent this, *ezDyson* does not compute cross sections for kinetic energies smaller than 0.1 eV when Coulomb waves are required. For more details, see the description of the `free_electron` section of the input in Section IV below.

More advanced users may be interested in running *ezDyson* with the Coulomb code that is available as part of the GNU Scientific Library (GSL) [21]. The GSL version of the Coulomb wave function is more efficient that the code currently used in *ezDyson*, and, moreover, does not suffer from problems for computing the Coulomb waves at low kinetic energies. Therefore, we also provide a makefile `mk.gsl` that can be used to compile *ezDyson* with GSL libraries. Note that the GSL libraries need to be installed prior to compilation. You may also need to modify the `mk.gsl` file to point to the correct "include" and "lib" directories. We also provide a modified `wavetypes` code that works with GSL to compute Coulomb and Bessel radial functions. To install *ezDyson* with GSL, execute these commands:

```
make clean
cp wavetypes.gsl.C wavetypes.C
cp wavetypes.gsl.h wavetypes.h
cp mk.gsl makefile
make
```

C. Running the program

The format for running an *ezDyson* calculation is:

```
exedys filename.xml >& filename.out &
```

To run the calculation in parallel, the `OMP_NUM_THREADS` environmental variable should be set equal to number of processors:

```
setenv OMP_NUM_THREADS 4
```

or, if using *bash*:

```
set OMP_NUM_THREADS=4
export OMP_NUM_THREADS
```

To run the calculation in serial, `OMP_NUM_THREADS` should be set to 1.

Consult the following section (Section IV) on how to generate the input *xml* file. For information on the output files generated and the information contained in them, see Section V.

IV. INPUT FILE

A. Generating input files

ezDyson input files are written in the *xml* format. *Python* scripts are available to automatically convert *Q-Chem*'s outputs with Dyson orbital calculations to an *ezDyson xml* input. However, details regarding the photoelectron experiment need to be specified manually by the user. The *Python* scripts are located in `/xmlsamples/makexml_scripts`. One script (`make_xml_IPEA.py`) works with EOM-IP or EOM-EA transitions computed with respect to the reference (CCSD) configuration. The second script (`make_xml_EE.py`) works with EOM-IP or EOM-EA transitions computed with respect to excited (EOM-EE) states. The third script (`make_xml_SF.py`) works with spin-flip transitions to/from IP/EA states. Please note that while the EE script is selective to EE transitions, the SF script includes all transitions including those from the high-spin reference. Finally, a fourth script, (`make_xml_molden.py`), may be used to get restricted Hartree-Fock (HF) or density functional theory Kohn-Sham (DFT KS) orbitals from Q-Chem's output files which were run with (`molden_format=true`).

The scripts can be used with the following command format:

```
make_xml_XX.py <filename.xml> <Q-Chem_dyson_job.out>
```

Once the input is ready, the user may run the program (see Section III). Always inspect the input to check for inconsistencies.

B. Input file structure

Here we explain the input structure and keywords for *ezDyson* by looking at a sample input file for calculating the photodetachment cross section from a fluoride anion. The sample *xml* input and the associated *Q-Chem* output file can be found in `/xmlsamples`. `fanion_default.xml` is the *xml* default file generated from the *Q-Chem* output `fanion.in.out` using the *Python* script, while `fanion.xml` is the same file already customized with details of the experiment, and ready to run.

```
------------------------------------------------
<?xml version="1.0" encoding="ISO-8859-1"?>

<root
  job = "dyson"
  >
------------------------------------------------
```

The first entry in the input specifies the job kind, and only has one only one option, `"dyson"`, so does not need to be modified.

```
--------------------------------------------------
<geometry
 n_of_atoms="1"
 text = "
     F          0.000000       0.000000       0.000000

  "
/>
--------------------------------------------------
```

The `<geometry>` field requires the number of atoms, followed by a list where each line contains the atom type and the Cartesian coordinates. If the input is prepared manually, make sure that the geometry in the *ezDyson* input is in the same orientation as in the electronic structure program that generated the Dyson orbitals.

```
----------------------------------------------------------------------------
<free_electron
    l_max = "3"
    charge_of_ionized_core = "0" >
        <k_grid n_points="3" min="0.01" max="2.01" />
</free_electron>
----------------------------------------------------------------------------
```

The `free_electron` field refers to the description of the photoelectron, and it requires the user to input the maximum angular momentum number, l, for which spherical waves $R_{kl}Y_{lm}$ will be generated and the dipole moment matrix elements calculated. Only functions with l up to 10 are implemented, but even this is excessive. To speed up the calculation, we recommend to use smaller `l_max`. As follows from the spherical harmonics expressions, the $+m$ and $-m$ components should always give identical values. For each l, all possible m projections are considered. The point group symmetry is not enforced, so if by symmetry certain m values should lead to zero transition dipoles, check for consistency in the output file. For atoms, one can save computing time by including only waves with l up to the angular momentum of the corresponding Dyson orbital plus 1, consistent with the selection rule. For instance, in the case of F^-/F where the Dyson orbital is of a p character (l=1), it is reasonable to use `l_max = "2"`.

The `charge_of_ionized_core` field refers to the charge of the system **after** photoejection of the electron. If it is set to zero, the photoelectron wave function is treated as a plane wave. This is the case for photodetachment, i.e., when the initial system is an anion and the ionized core is neutral. However, if `charge_of_ionized_core` is set to a non-zero positive integer, Coulomb waves are used instead of plane waves to account for the electrostatic interaction between the photoelectron and the ionized core of charge inputted in the field above. For photoionization of a neutral system, it is reasonable to use 1 for the charge of the ionized core [22].

Note: For larger molecules, using a Coulomb wave with charge of ionized core = 1 is not always the best option. In some cases, it is more reasonable to use plane waves even for photoionization where the actual charge of the core is 1. Also, it is possible to use a partial (effective) charge to describe a Coulomb wave that interacts more weakly with the core. It is not clear *apriori* what the best option would be for the description of the photoelectron: this topic is being studied, so it is left to the user to decide what to use for

`charge_of_ionized_core`, even if it is not the "actual" charge of the ionized core. See Ref. [22] for more details.

The next line contains the number of E_k's for which the cross section should be computed. On the same line the minimum and maximum E_k should be provided in eV, and the program generates an equally spaced number of energy points in the range provided. If the cross section is only needed at one E_k value, the user may specify `n_points = "1"`. In that case, the cross section will only be computed for the `min` E_k value.

Note that if `charge_of_ionized_core` is non-zero, *ezDyson* will not be able to compute cross sections for small kinetic energies. Therefore, it is recommended to use a minimum E_k of 0.1 eV or more. If a smaller E_k is used and `charge_of_ionized_core` is non-zero, a warning message will appear and the cross sections for these small E_k's will be set to zero automatically by the program. This problem can be avoided by using ezDyson with the GNU Scientific Library (see Section III).

```
-------------------------------------------------------
<averaging
   method= "avg"
   method_possible_values="noavg, avg, num" >
</averaging>
-------------------------------------------------------
```

In the `averaging` field the user may specify if they want the calculation in the molecular frame (noavg) or the lab frame (avg), or if they want to run the code without the plane wave expansion and therefore using numerical averaging (num). `"noavg"` computes the dipole matrix elements considering only one orientation of the molecule with respect to laser polarization, as both specified in the *ezDyson* input. `"avg"` computes the spherically averaged matrix elements and therefore spherically averaged cross sections, PADs, and anisotropy parameters (see Section II B). `"num"` runs the program without using the plane wave expansion, and uses numerical averaging using built-in orientations from the program REPULSION[19] (see Section II F). Note that if `averaging = "num"`, the `l_max` and `charge_of_ionized_core` values entered above are ignored. This approach has only been implemented for plane waves, not Coulomb waves.

```
-------------------------------------------------------
<laser
   ionization_energy = "3.40" >
   <laser_polarization x="0.0" y="0.0" z="1.0" />
```

```
</laser>
---------------------------------------------------
```

Next, information about the laser should be modified according to the experimental setup represented by the calculation. The ionization energy (IE) entry (in eV) is a property of the molecule/anion to be ionized (notice this is different from the energy of the ionizing radiation, which is the sum of the IE and E_k) . We recommend to use the best available IE (e.g., the experimentally determined value). If the absolute cross sections are not required (e.g., as in β anisotropy calculations), providing any number is good enough for running the program.

One should be careful in cross section calculations when several vibrational transitions are involved. A recommended option is to provide the ionization energy for the 0-0 transition and to recalculate the cross sections for other vibrational transitions by rescaling the 0-0 dipole moments elements, as described in Sections II D and V C.

The `laser_polarization` refers to the polarization of the laser with respect to the molecular axis in the input orientation. This field is only meaningful if no averaging is used (in the molecular frame). If the calculation is done in the lab frame, any information entered for the laser polarization is ignored. Note that PAD calculations in the molecular frame (noavg) are only correct if laser polarization is `x="0.0" y="0.0" z="1.0"`. If a different polarization is used, the PAD and β anisotropy calculations would be incorrect.

```
---------------------------------------------------
<lab_xyz_grid>
 <axis n_points="201" min="-10.0" max="10.0" />
</lab_xyz_grid>
---------------------------------------------------
```

`lab_xyz_grid` contains information about the dimensions of the xyz Cartesian grid (in Å) used for numerical integration to obtain C_{klm}, Eq. (14) in Section II. Usually a 0.1 Å (sometimes even 0.2 Å) spacing on each axis gives good results. Convergence with respect to the density of the xyz grid is quite readily achieved, although the user should check for the norm of the Dyson orbital to make sure the xyz box is large enough. The norm of the Dyson orbitals from *Q-Chem*, as integrated on the grid, should be very close to 1.0 since *Q-Chem* prints the coefficients of **normalized** Dyson orbitals. The norms are printed separately in the *Q-Chem* output, and are simply read from the*ezDyson* input (see below).

```
--------------------------------
<job_parameters
   unrestricted = "false"
   Dyson_MO_transitions = "1"
   spin_degeneracy = "2"
   orbital_degeneracy = "3"
   number_of_MOs_to_plot="0"
   MOs_to_plot = ""
/>
--------------------------------
```

<job_parameters> refers to details of the Dyson calculation by the *ab initio* program, e.g. *Q-Chem*. For both unrestricted and restricted HF calculations, two Dyson orbitals are read (alpha-left and alpha-right). Dyson_MO_transitions may be used to select which ionizing transition the *ezDyson* calculation needs to consider from the list of Dyson orbitals that follows below in the <dyson_molecular_orbitals> field. If the *Python* script was used to generate the *ezDyson* input, all Dyson orbitals would be copied into the *ezDyson* input but only the last transition is indicated in the Dyson_MO_transitions field. Therefore, it is necessary to specify which Dyson orbitals to use for the *ezDyson* calculation. For example, assume that the Dyson orbitals for photodetachment from the ground state of O^- to the ground state of O and to the first two excited states of O are calculated in *Q-Chem*. A total of 6 orbitals would be printed in the <dyson_molecular_orbital> section (two for each transition). Specifying Dyson_MO_transitions = "2" means that the 3rd and 4th orbitals are the ones to be used in the *ezDyson* calculation, corresponding to the ionization of O^- to the first excited state of O.

The spin_degeneracy and orbital_degeneracy fields are used to multiply the cross sections by the correct factor to account for spin and orbital degeneracies, as explained in Section II C. For photodetachment from F^-, there is a factor of two for spin degeneracy (one could ionize either the alpha or beta electron) and a factor of three for orbital degeneracy (there are three p orbitals from which ionization could take place).

If number_of_MOs_to_plot is different from 0, a mosplot.dat file is generated, containing the profile along the z molecular axis of the molecular orbitals specified in MOs_to_plot. Generally, this feature is useful if one wants to check for oscillatory behavior in the asymptotic region of the Dyson and/or HF orbitals. The left and right Dyson orbital profiles along x, y and z are printed by default in orbitalL.dat and orbitalL.dat, respectively.

`number_of_MOs_to_plot="0"` and `MOs_to_plot = ""` are the defaults and no MOs profiles are printed. The format for generating HF molecular orbitals profiles in `mosplot.dat` is for example, `number_of_MOs_to_plot="3"` and `MOs_to_plot = "5 6 9"`, for 5th, 6th and 9th MOs to be printed.

```
------------------------------------------------------------
<basis
  n_of_basis_functions="46"
  AO_ordering = "Q-Chem"
  purecart="                       111            (5D)">
  <atom
  text = "
F    0
S    8    1.000000
   1.95000000E+04     5.07000000E-04
   2.92300000E+03     3.92300000E-03
   6.64500000E+02     2.02000000E-02
   1.87500000E+02     7.90100000E-02
   6.06200000E+01     2.30439000E-01
   2.14200000E+01     4.32872000E-01
   7.95000000E+00     3.49964000E-01
   8.81500000E-01    -7.89200000E-03
S    8    1.000000
   1.95000000E+04    -1.17000000E-04
   2.92300000E+03    -9.12000000E-04
   6.64500000E+02    -4.71700000E-03
   1.87500000E+02    -1.90860000E-02
   6.06200000E+01    -5.96550000E-02
   2.14200000E+01    -1.40010000E-01
   7.95000000E+00    -1.76782000E-01
   8.81500000E-01     6.05043000E-01
S    1    1.000000
   2.25700000E+00     1.00000000E+00
S    1    1.000000
   3.04100000E-01     1.00000000E+00
P    3    1.000000
   4.38800000E+01     1.66650000E-02
   9.92600000E+00     1.04472000E-01
   2.93000000E+00     3.17260000E-01
P    1    1.000000
   9.13200000E-01     1.00000000E+00
P    1    1.000000
   2.67200000E-01     1.00000000E+00
D    1    1.000000
   3.10700000E+00     1.00000000E+00
D    1    1.000000
```

```
    8.55000000E-01    1.00000000E+00
F   1    1.000000
    1.91700000E+00    1.00000000E+00
S   1    1.000000
    9.15800000E-02    1.00000000E+00
P   1    1.000000
    7.36100000E-02    1.00000000E+00
D   1    1.000000
    2.92000000E-01    1.00000000E+00
F   1    1.000000
    7.24000000E-01    1.00000000E+00
****
       "
/>
</basis>
---------------------------------------------------------
```

The `<basis>` field contains information about the basis set used for computing the Dyson orbital. `n_of_basis_functions` requires the total number of AO basis functions for the molecule.

After that, the `AO_ordering` field can be used to indicate in which program/format the Dyson orbital coefficients were originally printed, since different programs sometimes print orbital coefficients in a different order. If the Dyson orbitals come from Q-Chem, then "Q-Chem" must be used for this field (this is the default). The second alternative is "Molden", if the Dyson orbitals were printed in Molden format.

Next, a 3-4 string of numbers is expected for `purecart`, specifying if pure (1, e.g. 5d) or Cartesian (2, e.g. 6d) Gaussians are used for $hgfd$ functions (or, if three numbers are provided, for gfd functions). Default is 1111. Note that *ezDyson* currently cannot process Dyson orbitals computed with basis sets that include h-functions and higher l.

Next, the program requires a list of the atoms and basis functions in the same order of atoms as the in the "geometry" field. The basis set is specified in the *Q-Chem* format:

X 0

L K scale

α_1 C_1^{Lmin} C_1^{Lmin+1}... C_1^{Lmax}

α_2 C_2^{Lmin} C_2^{Lmin+1}... C_2^{Lmax}

.

.

.

α_K C_K^{Lmin} C_K^{Lmin+1}... C_K^{Lmax}

***

where

X - Atomic symbol of the atom (atomic number is not accepted);

L - Angular momentum symbol (S, P, SP, D, F);

K - Degree of contraction of the shell (integer);

scale - Scaling to be applied to exponents (default is 1.00);

α_i - Gaussian primitive exponent (positive real number);

C_i^L - Contraction coefficient for each angular momentum (non-zero real numbers).

Basis sets for each atom are terminated with ****. No blank lines can be incorporated within the general basis set input. Note that more than one contraction coefficient per line is required for compound shells like SP.

```
--------------------------------------------------------------------------------
<!-- DMOs and MOs BELOW ARE FROM THE "fanion.in.out" Q-CHEM OUTPUT -->
<dyson_molecular_orbitals>
  <DMO   norm="0.9552"   transition="[REFERENCE|Mu|STATE  1/B1u ]"
comment="dyson left-right alpha"
    text="
   -8.3859134429139848e-18
    ...
   -2.6449557295922315e-09

      "  />
  <DMO   norm="0.9425"   transition="[REFERENCE|Mu|STATE  1/B1u ]"
comment="dyson right-left alpha"
    text="
   -7.3562475222956452e-18
    ...
   -1.0872190443650134e-09
    "  />

</dyson_molecular_orbitals>
--------------------------------------------------------------------------------
```

`<dyson_molecular_orbitals>` contains a list of the Dyson orbitals coefficients in AO basis in the order:

Transition1_left_alpha_Dyson

Transition1_right_alpha_Dyson

Transition2_left_alpha_Dyson

Transition2_right_alpha_Dyson

... etc.

At the top of each Dyson orbital the norm and two comments (`transition=""` and `comment=""`) are required.

```
--------------------------------------------------------------------
<molecular_orbitals total_number="46">
  <alpha_MOs>
    <MO                              type="alpha" number="1"   text="
    +5.47709686103895809772E-01
    ...
    " />
    ...
    <MO                              type="alpha" number="46"   text="
    -4.09334610487952866773E-16
    ...
    " />
  </alpha_MOs>

  <beta_MOs>
    <MO                              type="beta" number="1"   text="
    +5.47709686103895809772E-01
    ...
    " />
   ...
    <MO                              type="beta" number="46"   text="
    -4.09334610487952866773E-16
    ...
    " />
  </beta_MOs>

</molecular_orbitals>
--------------------------------------------------------------------
```

`<molecular_orbitals>` contains a list of all the alpha and beta molecular orbitals from the *Q-Chem* output. This field is not required if no MOs are required to be plotted in

<job_parameters> (in the sample input provided, `fanion.xml`, we replace this field with a filler that can generally be used if no MOs are required to be plotted but to prevent *ezDyson* from crashing).

```
<!-- Q-CHEM INPUT IS RESTORED FROM THE "fanion.in.out" Q-CHEM OUTPUT -->
<qchem_input>
---------------------------------------------------------------
$molecule
-1 1
F
$end

$rem
jobtype                   SP              single point
correlation               CCSD
basis                     aug-cc-pVTZ
purecart                  111             (5D)
unrestricted              false
cc_symmetry               false
eom_ip_states             [1]
cc_trans_prop             true
ccman2                    true
cc_do_dyson               true
cc_t_conv                 7
eom_davidson_convergence  6
print_general_basis       true
$end


----------------------------------------------------------------
</qchem_input>

</root>
-----------------------------------------------------------------------------
```

Finally, the *Q-Chem* input is copied at the end of the *xml* input for book-keeping purposes; it is not required for *ezDyson* to run.

V.   OUTPUT

A.   Main output

The first part of the output prints out detailed information of the input. This is useful for checking the input, for example, if the AO basis functions were assigned properly and

the correct columns of Dyson orbitals were read. For computational efficiency, the Dyson orbital coefficients below 1E-04 are treated as zero.

The first significant output information is the calculated norm of the Dyson orbital (left*left, right*right and left*right) on the xyz Cartesian grid. These are given as `Norm of the Dyson orbital integrated on the grid`. One should always check for convergence of the norm to 1.0 to make sure that the xyz box used is sufficiently large. In calculations of photoelectron matrix elements, the tail of the Dyson orbitals plays an important role, because spherical waves have significant density away from the core.

`Xavg Yavg Zavg` are the average values of the Cartesian coordinates over the left Dyson orbital, and they define the center of the lab and molecular axes. The center is shifted from the geometrical center to the center of the left Dyson orbital in order to ensure the orthogonality of the photoelectron wave function and Dyson orbital (Ref. [10]). The average square coordinates `<X^2> <Y^2> <Z^2> <R^2>` are useful for analyzing the size and shape of the left Dyson orbital.

Next, the square coefficients for each of the spherical waves considered are printed in the order of their wave vector magnitude k (in a.u.) and angular momentum l, m. After that, for each E_k a sum of all the square coefficients is printed.

Next, the PAD is calculated for `ntheta` number of θ angles (by default, `ntheta`=101 equally spaced points are taken from 0 to 2π) around the z lab axis, (see Section II E). `nkv` is the number of photoelectron k values for which the PAD is calculated, and is defined by the user in the `k_grid` section of the input. PADs are not printed in this output file, but are printed in a separate file named `pad.dat`. See section V B for more details.

The absolute total cross section at a certain E_k is computed using Eq. (17) and is printed next in the output. The format is:

```
E=IE+E_k,eV   xsec,a.u.
```

Similar to above, if the calculation is in the molecular frame the three Cartesian components of the cross sections are printed followed by the total cross section, while in lab frame calculation we print the cross section for x, y, and z-polarized light followed by the averaged cross sections.

In the next section, we provide the total cross sections (last column of previous table) divided by the ionizing radiation energy, E, listed by the energy of the photoelectron E_k in

eV. This format is more useful for accounting for vibrational transitions using the `xsecFCF` script, as explained in Section V C (see also Ref. [10]).

β anisotropy parameters are computed using spherical harmonics $Y_{lm}^*(\hat{k})Y_{lm}(\hat{k})$ (defined with the preferential axis along z in the case the calculation is in the molecular frame).

The columns in this part of the output correspond to:

```
E=IE+E_k,eV    Sigma_par      Sigma_perp     Sigma_tot      beta
```

`sigma_tot` here should be the same as the total cross section printed above under the heading `xsec, a.u.`

The *ezDyson* input file is printed at the end of the output.

B.   Additional output files

`orbitalL.dat` and `orbitalR.dat` contain the profiles of the left and right alpha Dyson orbital along the three axes as follows:

```
x  L(D)Dys(x,0,0)  L(R)Dys(x,0,0)*x  y  L(R)Dys(0,y,0)  L(R)Dys(0,y,0)*y  z
L(R)Dys(0,0,z)  L(R)Dys(0,0,z)*z  ...
```

`pad.dat` contains the PAD calculated using spherical harmonics $Y_{lm}(\hat{k})$ with z polarized light in the format:

```
azimuthal_angle,radians    PAD_sq_contrib(lm)    PAD_cross_contrib(lml'm)
total_PAD
```

`mosplot.dat` is generated only if MO profiles were requested in the input `<job_parameters>`. The format is:

```
z  MOi(0,0,z)  MOi(0,0,z)*z  MOj(0,0,z)  MOj(0,0,z)*z  ...
```

All quantities in these outputs are in atomic units, unless specified otherwise.

C.   Script for accounting for Franck-Condon factors

After computing total cross sections in *ezDyson*, it is possible to account for vibrational wave functions, as discussed in Section II D, by computing Franck-Condon factors

(FCFs). This is done by a separate program, *ezSpectrum*, which is available for download at `iopenshell.usc.edu/downloads/`. Cross-sections accounting for FCFs can be then computed using Eq. (36). However, Eq. (36) requires summing over D_k^{IF} for different vibronic transitions. Since *ezDyson* only computes cross sections at discrete values of E_k, we should integrate D_k^{IF} over a range of E_k values, or, equivalently, over a range of photon energies E, for each discrete step. Instead, we use a linear interpolation to approximate the value of D_k^{IF} for each vibronic transition at certain discrete values of E, which makes it possible to sum these terms. This is done automatically by a separate script (*xsecFCF*) which requires data given in the outputs of *ezDyson* and *ezSpectrum*.

The script can be found in in `xmlsamples/xsecFCF`. It requires two files called `FCFs.dat` and `xsec.dat`, as shown in the same folder. The first file is simply an output file from *ezSpectrum*, usually called `filename.xml.spectrum_parallel` or `filename.xml.spectrum_duschinsky` (depending on the options used in *ezSpectrum*). This file needs to be modified by adding only one line in the beginning of the file indicating the total number of FCF transitions included in the file. The second file, `xsec.dat`, is just a segment copied from the *ezDyson* output from the section titled "`Total cross-section / E (in a.u.) vs. Ek (in eV)`", with again one additional line indicating the number of energies for which cross sections were computed. The ionization energy is read from `FCFs.dat`, so it is important that the correct ionization energy is entered in the *ezSpectrum* input file. Finally, please note that this script can only be used for a vibrationally cold calculation (i.e., using low temperatures in *ezSpectrum*). The script will not work correctly if the ground state is also vibrationally excited. The script needs to be run in the same folder as `FCFs.dat` and `xsec.dat`. To run the script, use the following command:

```
./xsecFCF
```

[1] A.I. Krylov, Equation-of-motion coupled-cluster methods for open-shell and electronically excited species: The hitchhiker's guide to Fock space, Annu. Rev. Phys. Chem. **59**, 433 (2008).

[2] Y. Shao, Z. Gan, E. Epifanovsky, A.T.B. Gilbert, M. Wormit, J. Kussmann, A.W. Lange, A. Behn, J. Deng, X. Feng, D. Ghosh, M. Goldey, P.R. Horn, L.D. Jacobson, I. Kaliman, R.Z. Khaliullin, T. Kus, A. Landau, J. Liu, E.I. Proynov, Y.M. Rhee, R.M. Richard, M.A.

Rohrdanz, R.P. Steele, E.J. Sundstrom, H.L. Woodcock III, P.M. Zimmerman, D. Zuev, B. Albrecht, E. Alguires, B. Austin, G.J.O. Beran, Y.A. Bernard, E. Berquist, K. Brandhorst, K.B. Bravaya, S.T. Brown, D. Casanova, C.-M. Chang, Y. Chen, S.H. Chien, K.D. Closser, D.L. Crittenden, M. Diedenhofen, R.A. DiStasio Jr., H. Do, A.D. Dutoi, R.G. Edgar, S. Fatehi, L. Fusti-Molnar, A. Ghysels, A. Golubeva-Zadorozhnaya, J. Gomes, M.W.D. Hanson-Heine, P.H.P. Harbach, A.W. Hauser, E.G. Hohenstein, Z.C. Holden, T.-C. Jagau, H. Ji, B. Kaduk, K. Khistyaev, J. Kim, J. Kim, R.A. King, P. Klunzinger, D. Kosenkov, T. Kowalczyk, C.M. Krauter, K.U. Laog, A. Laurent, K.V. Lawler, S.V. Levchenko, C.Y. Lin, F. Liu, E. Livshits, R.C. Lochan, A. Luenser, P. Manohar, S.F. Manzer, S.-P. Mao, N. Mardirossian, A.V. Marenich, S.A. Maurer, N.J. Mayhall, C.M. Oana, R. Olivares-Amaya, D.P. O'Neill, J.A. Parkhill, T.M. Perrine, R. Peverati, P.A. Pieniazek, A. Prociuk, D.R. Rehn, E. Rosta, N.J. Russ, N. Sergueev, S.M. Sharada, S. Sharmaa, D.W. Small, A. Sodt, T. Stein, D. Stuck, Y.-C. Su, A.J.W. Thom, T. Tsuchimochi, L. Vogt, O. Vydrov, T. Wang, M.A. Watson, J. Wenzel, A. White, C.F. Williams, V. Vanovschi, S. Yeganeh, S.R. Yost, Z.-Q. You, I.Y. Zhang, X. Zhang, Y. Zhou, B.R. Brooks, G.K.L. Chan, D.M. Chipman, C.J. Cramer, W.A. Goddard III, M.S. Gordon, W.J. Hehre, A. Klamt, H.F. Schaefer III, M.W. Schmidt, C.D. Sherrill, D.G. Truhlar, A. Warshel, X. Xu, A. Aspuru-Guzik, R. Baer, A.T. Bell, N.A. Besley, J.-D. Chai, A. Dreuw, B.D. Dunietz, T.R. Furlani, S.R. Gwaltney, C.-P. Hsu, Y. Jung, J. Kong, D.S. Lambrecht, W.Z. Liang, C. Ochsenfeld, V.A. Rassolov, L.V. Slipchenko, J.E. Subotnik, T. Van Voorhis, J.M. Herbert, A.I. Krylov, P.M.W. Gill, and M. Head-Gordon, Advances in molecular quantum chemistry contained in the Q-Chem 4 program package, Mol. Phys. **113**, 184 (2015).

[3] http://www.q-chem.com.

[4] H.A. Bethe and E.E. Salpeter, *Quantum mechanics of one and two electron atoms*. Plenum, New York, 1977.

[5] K.J. Reed, A.H. Zimmerman, H.C. Andersen, and J.I. Brauman, Cross sections for photodetachment of electrons from negative ions near threshold, J. Chem. Phys. **64**, 1368 (1976).

[6] C.M. Oana and A.I. Krylov, Dyson orbitals for ionization from the ground and electronically excited states within equation-of-motion coupled-cluster formalism: Theory, implementation, and examples, J. Chem. Phys. **127**, 234106 (2007).

[7] D.M. Gingrich, *Practical Quantum Electrodynamics*. CRC Press, 2006.

[8] J.J. Sakurai, *Modern Quantum Mechanics*. Addison-Wesley Publishing Company, 1995.

[9] L.D. Landau and E.M. Lifshitz, *Quantum Mechanics: Non-relativistic theory*. Pergamon, Oxford, 1977.

[10] C.M. Oana and A.I. Krylov, Cross sections and photoelectron angular distributions in photodetachment from negative ions using equation-of-motion coupled-cluster Dyson orbitals, J. Chem. Phys. **131**, 124114 (2009).

[11] S.S. Andrews, Using rotational averaging to calculate the bulk response of isotropic and anisotropic samples from molecular parameters, J. Chem. Educ. **81**, 877 (2004).

[12] Y. Lin and C. Ning, Calculation of photodetachment cross sections and photoelectron angular distributions of negative ions using density functional theory, J. Chem. Phys. **143**, 144310 (2015).

[13] E.P. Wigner, *Group Theory*. New York: Academic Press, 1959.

[14] F.O. Ellison, Theoretical equations for photoionization cross sections of polyatomic molecules in planewave and orthogonalized planewave approximations, J. Chem. Phys. **61**, 507 (1974).

[15] M.E. Rose, *Elementary Theory of Angular Momentum*. John Wiley and Sons, New York,

1957.

[16] K. Schulten and R. G. Gordon, Recursive evaluation of 3j and 6j coefficients, Comp. Phys. Comm. **11**, 269278 (1976).

[17] K. Schulten, Exact recursive evaluation of 3j- and 6j-coefficients for quantum-mechanical coupling of angular momenta, J. Math. Phys. **16**, 1961 (1975).

[18] J. Luscombe and M. Luban, Simplified recursive algorithm for wigner 3j and 6j symbols, Phys. Rev. E **57**, 72747277 (1998).

[19] M. Bak and N.C. Nielsen, Repulsion, a novel approach to efficient powder averaging in solid-state nmr, J. Magn. Reson. **125**, 132 (1997).

[20] Thomas Smid, Program code for regular coulomb wave function, http://www.plasmaphysics.org.uk/programs/coulombc.htm, 2007, (Accessed August 13, 2015).

[21] http://www.gnu.org/software/gsl/.

[22] S. Gozem, A.O. Gunina, T. Ichino, D.L. Osborn, J.F. Stanton, and A.I. Krylov, Photoelectron wave function in photoionization: Plane wave or Coulomb wave?, J. Phys. Chem. Lett. **6**, 4532 (2015).

[23] Note that FCF sometimes refers to the square of the Franck-Condon overlap integral. In *ezSpectrum*, FCF refers to the overlap integral and the square of it is called 'Intensity'.