SUPPORTING INFORMATION
iSPECTRON: a simulation interface for linear and nonlinear spectra with ab-initio quantum chemistry software

Francesco Segatta[1], Artur Nenov[1], Daniel R. Nascimento[2], Niranjan Govind[2,*], Shaul Mukamel[3,*], and Marco Garavelli[1,*]

[1]Dipartimento di Chimica Industriale "Toso Montanari", Università degli Studi di Bologna, Viale del Risorgimento 4, I-40136 Bologna, Italy
[2]Physical and Computational Sciences Directorate, Pacific Northwest National Laboratory, Richland, Washington 99352, United States
[3]Department of Chemistry and Department of Physics and Astronomy, University of California, Irvine, 92697, USA

September 24, 2020

1 Auxiliary communication files format

iSPECTRON gives the possibility to avoid reading the QC software outputs by providing some (or all of) the information via auxiliary communication files. To be recognized as such, these should have a well defined format, including (i) a header, which permits iSPECTRON to recognize the data type stored in the corresponding file, and (ii) the data itself, structured according to the specified data type. In particular, one may pass:

- **The list of vertical transition energies** with respect to the GS (0 by definition), given as a single column file, in increasing order and including the energy of the GS, in cm^{-1} units, with header

    #free format - ener(gies)

    and the part within the brackets is optional.

- **The transition dipole moments** between the various states, given as a five column file in which the first two indices represent initial and final states a and b (GS has index 0), while the last three numbers are the x, y and z components of the transition dipole $\boldsymbol{\mu}_{ab}$ in atomic units. The file content should be preceded by the header

    #free format - dipo(les)

- **The vertical gradients**, given as four column files (one for each state of interest), in which the first column is the atom type, while the last three numbers are the x, y and z components of the gradient (in Hartree/Bohr units). The header should not only specify the information content, but also the index i of the state to which the gradient refers to (the GS has index 1):

    #free format - grad(ient)
    #gradient root i

    The total number of lines - not counting the header - should match the total number of atoms.

In case the user wants to provide one (or several) of the above defined auxiliary files, the keyword $-free$, followed by the total number of states (including GS), should be used.

2 QC data for linear absorption

We report here the parameters that were used for the QC calculation and for the linear absorption spectra simulations reported in Figure 5 of the main text. The Overdamped brownian oscillator (OBO) model parameters for the system-bath spectral densities were set, at both levels of theory, to $\lambda_b = 150$ cm^{-1} and $\Lambda_b = 80$ cm^{-1} at $T = 300K$ (while $\lambda_b = 8$ cm^{-1} and $\Lambda_b = 80$ cm^{-1} at $T = 5K$). The vertical and adiabatic energy gaps (ΔE_V and ΔE, respectively), the oscillator strength, as well as the total intra-molecular reorganization

energy (λ) for the La state, are reported in Table 1. The La spectral densities are reported in Figure 1.

RASSCF/RASPT2 computations were performed with OpenMolcas, on the top of a MP2 optimized pyrene structure, employing an active space defined as RAS(8,4|0,0|7,4).[1] A state average of eight roots was chosen, and the La state was identified (by its configuration and its intense transition dipole moment) as root number 5 at RASSCF level. The energies of the GS and of the La state were corrected at the single state (SS)-RASPT2 level. The vertical energy gradient of La was computed numerically at the SS-CASPT2 level. Normal modes and frequencies in the GS were obtained at the MP2 level. The ANO-L-VDZP basis set was used throughout.

TDDFT computations were performed with NWChem: first a GS optimization was accomplished, followed by the evaluation of the GS normal modes and frequencies with *pbe0* exchange-correlation functional, and the 6-31G* basis set. Excitation energies, vertical gradients and transition dipole moments were computed for two excited states at the TDDFT level (with the same functional and basis set specified above), and the La was identified as the first excited state.

	RASSCF/PT2				TDDFT			
	ΔE_V	λ	ΔE	TDM^2	ΔE_V	λ	ΔE	TDM^2
La	28670	1260	27400 (365 nm)	2.76	30890	1190	29700 (337 nm)	2.93

Table 1: ΔE_V, ΔE and λ are given in cm^{-1} (if not otherwise specified), while TDM^2 is reported in au^2. Note that $\Delta E = \Delta E_V - \lambda$.

3 QC data for nonlinear spectroscopy

The QC data for nonlinear spectroscopy (Figure 6 of the main text) were produced only at the RASSCF/RASPT2 level of theory, by means of the OpenMolcas software. These computations were performed by some of the authors in a previous publication, and the interested reader may look at Ref. [1] for more details. We only report a few relevant details here:

- The computations are performed with D_{2h} symmetry;

- The states from the f manifold coupled through a non-zero transition dipole moment to either La or Lb are computed at the geometry of the respective excited state minimum. The vertical gradients are transformed accordingly to account for the different reference geometries;

---

[1]Here the first two indices refer to the number of doubly occupied active orbitals in RAS1 and the number of possible holes that may be created in that space, respectively. The following two numbers are the number of active electrons and the number of active orbitals in the RAS2 space. Finally, the last two numbers refer to the number of virtual orbitals in RAS3 and the number of possible electrons that can be put in that space, respectively.
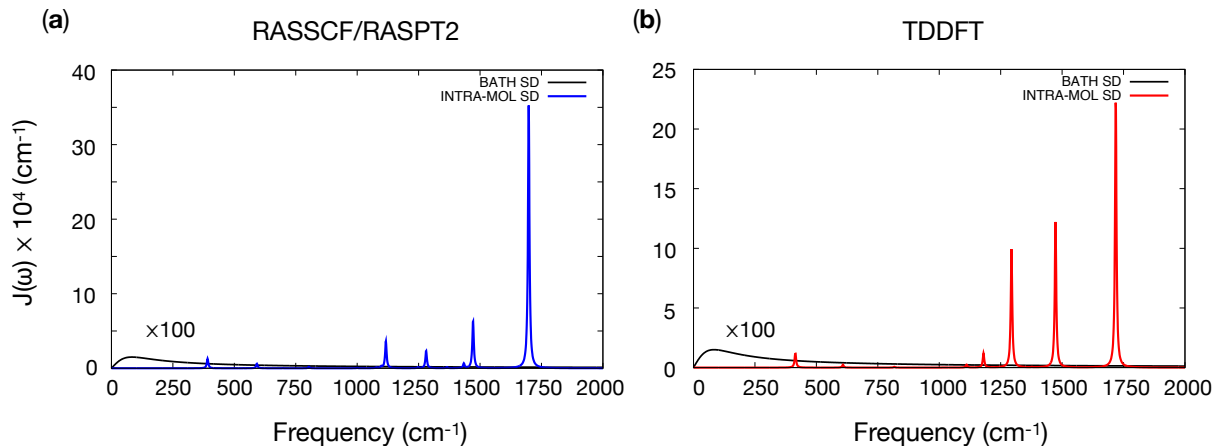
(a) RASSCF/RASPT2

(b) TDDFT

Figure 1: Spectral densities at the (a) RASSCF/RASPT2 and (b) TDDFT level of theory for the La state, drawn with blue and red lines, respectively. The OBO spectral density (black curve) was multiplied by a factor 100 to make it visible in the spectra.

- As version 1.0 of iSPECTRON is not designed to process the output of computations performed with symmetry, the gradient information are provided manually by means of the auxiliary communication files discussed in Section 1.

The OBO parameters are the same of the linear absorption at 300K. Transition energies, oscillator strengths and the total intra-molecular reorganization energy (λ) for every electronic state in the studied energy window, are reported in Tables 2 and 3. The content of the communication files passed to the interface are reported in Section 1.

		e manifold		
	ΔE_V	λ	ΔE	TDM^2
Lb	25590	660	24930 (400 nm)	0
La	30140	810	29330 (340 nm)	4.31

Table 2: ΔE_V, ΔE and λ are given in cm^{-1}, while TDM^2 is reported in au^2. Note that the use of symmetry slightly affects the vertical transition energy, reorganization energy and dipole moment magnitude of the La state with respect to the values obtained without symmetry (Table 1).

		f manifold			
initial st.	*final st.*	$\Delta E_V^{(a)}$	λ	$\Delta E^{(b)}$	TDM^2
Lb	S3	45030	620	19480 (513 nm)	0.64
Lb	S4	46370	760	20680 (483 nm)	0.67
La	S5$^{(c)}$	46890	1140	16420 (610 nm)	3.90
La	S6	47070	8150	9590 (1040 nm)	3.50
Lb	S7	53080	1040	27110 (370 nm)	2.35
La	S8	53800	860	23610 (423 nm)	0.50
La	S9	55700	1820	24550 (407 nm)	0.95

Table 3: ΔE_V, ΔE and λ are given in cm^{-1}, while TDM^2 is reported in au^2. [a] vertical excitation energy with the GS as reference; [b] adiabatic excitation energy of the final state with the initial state as reference; the total (intra-molecular) reorganization energy for each state is given with respect to the GS minimum. [c] The computed value was shifted by $\sim$3000 cm^{-1} in the blue to match the ESA experimental position.

4   Input files for nonlinear spectroscopy

In Figure 2 we report a sample of the iSPECTRON communication files containing the QM data for nonlinear spectroscopy. The energies are listed in the *energies.txt* file, the transition dipole moments in *tdms.txt*, while the gradients for the *Lb* and *La* states in the *S1_grad.xyz* and *S2_grad.xyz* files, respectively (see Section 5).


```
free format - energ
0
25593.41
30138.21
44889.60
45029.28
46366.88
47071.35
53082.51
53796.50
55700.86



free format - dipoles
0 2 -2.0776   0.0000   0.0000
2 3  1.9749   0.0000   0.0000
2 6  1.8696   0.0000   0.0000
2 8  0.0000  -0.7077   0.0000
2 9 -0.9729   0.0000   0.0000
1 4  0.0000  -0.7981   0.0000
1 5  0.8212   0.0000   0.0000
1 7 -1.5332   0.0000   0.0000
```

```
free format - grads          → Lb
gradient root 2
C  -0.012920   0.002776   0.000000
C  -0.001325  -0.014989   0.000000
C   0.031276   0.000000   0.000000
C   0.001004  -0.011845   0.000000
C  -0.016406   0.000000   0.000000
H  -0.000007   0.000817   0.000000
H   0.000445   0.000610   0.000000
H   0.000874   0.000000   0.000000
C   0.012920   0.002776   0.000000
C   0.001325  -0.014989   0.000000
C  -0.001004  -0.011845   0.000000
H   0.000007   0.000817   0.000000
H  -0.000445   0.000610   0.000000
C   0.012920  -0.002776   0.000000
C   0.001325   0.014989   0.000000
C  -0.031276   0.000000   0.000000
C  -0.001004   0.011845   0.000000
C   0.016406   0.000000   0.000000
H   0.000007  -0.000817   0.000000
H  -0.000445  -0.000610   0.000000
H   0.000874   0.000000   0.000000
C  -0.012920  -0.002776   0.000000
C  -0.001325   0.014989   0.000000
C   0.001004   0.011845   0.000000
H  -0.000007  -0.000817   0.000000
H   0.000445  -0.000610   0.000000
```

```
free format - grads          → La
gradient root 3
C  -0.027601   0.008895   0.000000
C   0.019638  -0.011786   0.000000
C   0.006750   0.000000   0.000000
C  -0.016556  -0.000792   0.000000
C  -0.003301   0.000000   0.000000
H  -0.000327   0.000651   0.000000
H  -0.000107   0.000227   0.000000
H   0.000486   0.000000   0.000000
C   0.027601   0.008895   0.000000
C  -0.019638  -0.011786   0.000000
C   0.016556  -0.000792   0.000000
H   0.000327   0.000651   0.000000
H   0.000107   0.000227   0.000000
C   0.027601  -0.008895   0.000000
C  -0.019638   0.011786   0.000000
C  -0.006750   0.000000   0.000000
C   0.016556   0.000792   0.000000
C   0.003301   0.000000   0.000000
H   0.000327  -0.000651   0.000000
H   0.000107  -0.000227   0.000000
H  -0.000486   0.000000   0.000000
C  -0.027601  -0.008895   0.000000
C   0.019638   0.011786   0.000000
C  -0.016556   0.000792   0.000000
H  -0.000327  -0.000651   0.000000
H  -0.000107  -0.000227   0.000000
```

Figure 2: Sample of the iSPECTRON communication files used to generate the nonlinear spectra.

5 iSPECTRON command line input examples

Linear absorption: The Spectron input file for the linear absorption computation taking the data from NWChem are produced by running the following line:

> python3 iSPECTRONa.py pyrene_root1_grad.out pyrene_freq.out -w1i 26000 -w1f 36000 -nw1 1000 -sig LA

where *-sig LA* denotes the selected signal type (linear absorption), *pyrene_root1_grad.out* and *pyrene_freq.out* are the NWChem output files containing energy, transition dipole and gradient of the *La* state (first file), as well as GS normal modes and frequencies (second file). The remaining options specify the window range in cm^{-1} (*-w1i* and *-w1f*, where i and f stand for initial and final, respectively) and the number of samples in the specified window (*-nw1*).

Nonlinear spectroscopy: QC input for nonlinear spectroscopy simulation is obtained via user specified communication files containing energies, transition dipole moments and gradients. GS frequencies are also provided. The following line of commands is executed

> python3 iSPECTRONa.py -free 10 energies.txt tdms.txt geometry.log S2_grad.xyz S1_grad.xyz S3_grad.xyz S4_grad.xyz S5_grad.xyz S6_grad.xyz S7_grad.xyz S8_grad.xyz S9_grad.xyz -sig PP -w1i 10000 -w1f 40000 -w3i 10000 -w3f 40000 -nw1 300 -nw3 300

where *-free 10* identifies that the input files have been prepared by the user (with the formats specified in Sec. 1, and a total of 10 states (including the GS) are to be considered. Energies are provided in the *energies.txt* file, transition dipole moments in the *tdms.txt* file, while frequencies and normal modes are given in the *geometry.log* file (produced by the COBRAMM software[2]). The following files (with *.xyz* extension) specify the vertical gradients (referred to the GS minimum) for all the sates of interest. Finally, *-sig PP* specifies the type of signal (third-order nonlinear signal), while the remaining commands specify the window range in cm^{-1} and the number of samples.

Data analysis of the spectra generated by Spectron is facilitated by the second leg of the interface. In particular, the various spectra reported in Figure 6 of the main text were obtained with the following commands:

- **Pump-probe (infinite time-resolution):**

> python3 iSPECTRONb.py PPfromS2andS1modESA/ -sig2plot ppheatmap -t2i 0 -t2f 500 -units nm fs -w1i 310 -w1f 350 -w3i 300 -w3f 650 -sigmat2 0

where PPfromS2andS1 is the parent directory, which contains 300 t_2 sub-folders with 2D maps (one every 2 fs from 0 to 600 fs), *-sig2plot* allows to select the signal type, here the 1D pump probe heat map (*ppheatmap*); *-units* allows to select the units for time and energy (*nm* and *fs* in this case); the other options specify the t_2 time range in fs (*-t2i* and *-t2f*, where i and f stand for initial and final, respectively), the pump (*-w1i\-w1f*) and probe (*-w3i\-w3f*) spectral windows in the specified units (nm); finally, the *-sigmat2* option allows to consider the finite time resolution of the experiment, which

is given here as the standard deviation (in fs) of the Gaussian that convolutes the data over the t_2 time. By setting this value to zero, we are considering an infinite (time) resolution experiment (i.e. the "impulsive limit").

- **Fourier transform of Pump-probe (finite time-resolution):**

  python3 iSPECTRONb.py PPfromS2andS1modESA-last-nolifetime/ -sig2plot ftppheatmap -t2i 0 -t2f 600 -w1i 300 -w1f 450 -w3i 550 -w3f 650 -sigmat2 7.5 --silence

  where the signal type is Fourier transform analysis of the 1D pump probe map (*ftppheatmap*)[2] and *-sigmat2* is set to 7.5 fs. The keyword (*--silence*) is used to avoid direct plotting of the data at the end of the iSPECTRON execution (the *gnuplot* file will be produced but not executed).

- **2DES:**

  python3 iSPECTRONb.py PPfromS2andS1modESA-last/ -sig2plot 2dmap -t2i 50 -t2f 50 -units nm fs -w1i 330 -w1f 355 -w3i 330 -w3f 355 -sigmat2 7.5 –silence

  where the signal type is a 2DES map (*2dmap*) generated at time $t_2 = 50$ fs (selected through the *-t2i\-t2f* specifiers).[3]

- **Point/Region t_2 time trace:**

  python3 iSPECTRONb.py PPfromS2andS1modESA-spaced/ -sig2plot point2 -t2i 0 -t2f 4000 -units nm fs -w1i 340 -w1f 345 -w3i 610 -w3f 620 -sigmat2 7.5

  where the signal type is a t_2 time trace (*-sig2plot point2*) over a period of 4000 fs defined by the *-t1i\-t2f* specifiers; the spectral window specifiers *-w1i\-w1f* and *-w3i\-w3f* are here used to identify the region whose average value has to be traced during the time t_2.

- **Spectral diffusion:**

  python3 iSPECTRONb.py PPfromS2andS1modESA-spaced -sig2plot SpecDiff2 -t2i 0 -t2f 4000 -units nm fs -w1i 330 -w1f 355 -sigmat2 7.5 --silence

  where the signal type is spectral diffusion via calculation of the flattening parameter (*-sig2plot SpecDiff2*) over a period $t_2 = 4000$ fs defined by the *-t1i\-t2f* specifiers; the spectral window is selected by the *-w1i\-w1f* and *-w3i\-w3f* specifiers in order to isolate the GSB peak around 340 nm[4].

---

[2]Note that the data provided to the interface has to be manipulated to remove the decaying part of the signal, a necessary post-processing of the data prior to the Fourier transform analysis.

[3]Note that, also the data before and after $t_2 = 50$ fs will be used if present, in order to implement the convolution specified by the *-sigmat2* keyword.

[4]Note that if the set of *-w3i\-w3f* specifiers is not defined, a symmetric region is automatically selected.

References

[1] A. Picchiotti, A. Nenov, A. Giussani, V. I. Prokhorenko, R. J. D. Miller, S. Mukamel, and M. Garavelli. Pyrene, a test case for deep-ultraviolet molecular photophysics. *The Journal of Physical Chemistry Letters*, 10(12):3481–3487, May 2019.

[2] O. Weingart, A. Nenov, P. Altoè, I. Rivalta, J. Segarra-Martí, I. Dokukina, and M. Garavelli. COBRAMM 2.0 — a software interface for tailoring molecular electronic structure calculations and running nanoscale (QM/MM) simulations. *Journal of Molecular Modeling*, 24(9), Sept. 2018.