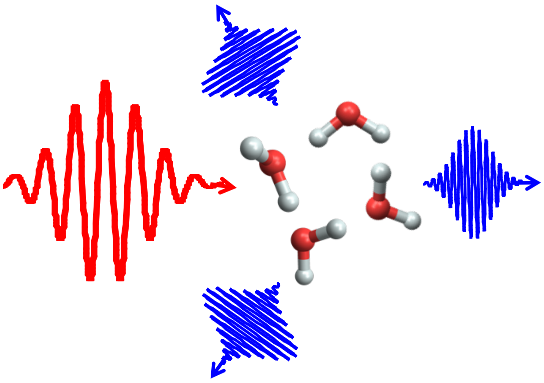
# PySHS V2.2 : a Computational Second Harmonic Scattering Code



User Manual 2.1 Written by:

Pierre-Marie Gassin

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

This user manual is distributed along with the pySHS software to aid in setting up the input files required for carrying out a Second Harmonic Scattering simulation. Every effort is made to release the most updated and complete version of the manual. To report any inconsistencies, errors or missing information, or to suggest improvements, send email to pierre-marie.gassin@enscm.fr.

# Acknowledgements

Support for this work was provided by a grant from CAMOMILS Research National French Agency :  ANR-15-CE21-0002.

We would like to acknowledge the Prof. Pierre-François Brevet for the numerous stimulating discussions about Second Harmonic Scattering and  Dr Julien Duboisset for its HRS_computing program which can be seen as a precursor to PySHS.

The people who have contributed to the development of the code include:

- Dr Pierre-Marie Gassin
- Dr Lotfi Boudjema
- Dr Gaelle Gassin

# Contents

Chapter 1: Introduction

Chapter 2: General Equations Implemented in PySHS

Chapter 3: Incoherent Second Harmonic Scattering : HRS program

Chapter 4: Coherent Second Harmonic Scattering of Supramolecular structures : SHS program

Chapter 5: Coherent Second Harmonic Scattering  of Colloidal object : Sphere program

Chapter 6 :  Additional Scripts and Perspectives

Chapter1

# Introduction

## 1) What is new in PySHS V2.2?

PySHS V1 was a collection of different programs written in python which can compute the Second Harmonic Intensity Scattered by different class of objects. For large system with numerous correlated dipoles (N>1000) , the calculation in PySHS V1.0 can be very long because of the intrinsic limitation of the Python software. The new version, PySHS V2.2, has been entirely rewritten in C and it increases greatly the speed of the calculation.

The others modifications of the V2.2 code concern its ability to take imaginary refractive index for the medium as requested by some users and also to compute polarization resolved SHS for any scattering angle (not only right angle and transmission collection). The definition of the Euler angle and the transformation matrix has been changed in order to increase the clarity and to avoid confusion. To summarize this software, PySHS V2.2 is able to compute the Second Harmonic Intensity Scattered by different class of objects:

- incoherent molecules in solution (see program HRS)
- supramolecular structures formed by correlated molecules (see program SHS).
- colloidal sphere in solution. (see program sphere_SHS)
- colloidal cylinder in solution. (see program cylinder_SHS, coming soon in the V2.2.1)

## 2) Distribution

PySHS has been written in C. So, a compilation step is required (see the following paragraph). After unpacking the pySHS distribution, the parent PySHS directory will contain the following subdirectories:

- Documentation

It contains the user_guide. You may find most answers to your questions here.

- Examples

Contains some examples of calculation and in particular the input and output files.

- Src

Contains the pySHS source code and some script used to generate the input files.

- Work

It is the directory where the user can store the input and output file.

# 3) Compilation and Run

A C compiler has to be installed (on macOS or linux, gcc is normally already install, to check it, write in a command line: gcc --version ). To compile PySHS under MacOS or linux, enter in a terminal:

```
>gcc -o Work/SHS Src/SHS.c -lm
```

This command will create the SHS executable file in the Work directory.
Do the same things for the sphere_SHS program and for th HRS_program:

```
>gcc -o Work/sphere_SHS Src/sphere_SHS.c -lm
>gcc -o Work/HRS Src/HRS.c -lm
```

Running a pySHS calculation
---------------------------

Depending of the pySHS program used, it requires one (sphere_SHS or HRS programs) or two input files (SHS program) in order to run.

to launch, the SHS program, in the PySHS_V2 directory enter:

```
>./Work/SHS <Keyword>  <inputbeta>  <input orientation>   <outputfile>
```

where <Keyword> is equal to “polarplot_single” or “polarplot_integrate” or “angle_scattering” depending on the computation you want to perform. <inputbeta>, <input_orientation> and <outputfile> are respectively the name of the input file containing the hyperpolarizability of the molecule, the name of the input file containing the orientation and position of the molecules and the output file.

to launch, the sphere_SHS program, enter:

```
>./Work/sphere_SHS <Keyword> <inputbeta> <outputfile>
```

where <Keyword> is equal to “polarplot_single” or “polarplot_integrate” or “angle_scattering” depending on the computation you want to perform. inputbeta, and outputfile are respectively the name of the input file containing the hyperpolarizability of the molecule, and the output file.

Another output files formatted to be plotted with gnuplot (http://www.gnuplot.info) are also generated: out_plot and out_arrows_gnuplot.

# Chapter 2

# General Equations Implemented in PySHS

## 1) The Notation

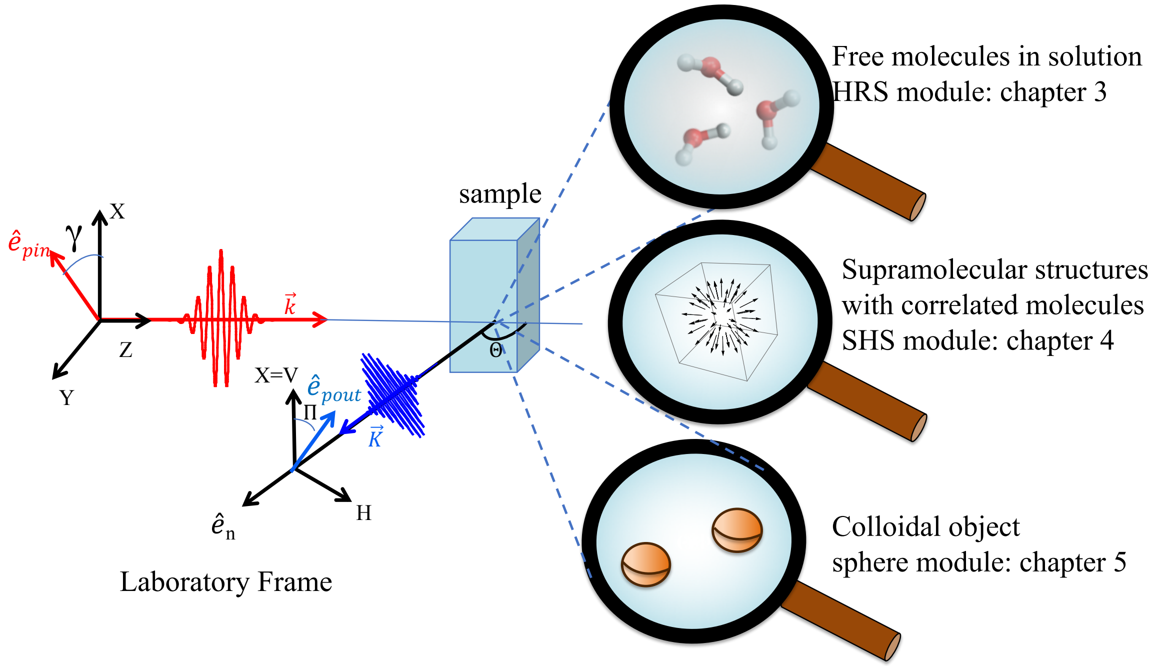


Figure 1: definition and notation

The fundamental light at the $\omega$ frequency is defined by its wave vector $\vec{k} = k\hat{e}_z$ and its polarization direction $\hat{e}_{p-in}$. The input polarization angle $\gamma$ is equal to the angle between $(\hat{e}_X, \hat{e}_{p-in})$

The second harmonic scattered light at the $2\omega$ frequency is defined by its wave vector $\vec{K} = K\hat{e}_n$ and the angle $(\hat{e}_z, \hat{e}_n)$ defines the scattered angle $\Theta$. The polarization of the second harmonic scattered light is defined by $\hat{e}_{p-out}$ and the output polarization angle $\Pi$ is equal to the angle between $(\hat{e}_X, \hat{e}_{p-out})$. In the laboratory frame, we have the following relation:

$$\hat{e}_n = \begin{pmatrix} 0 \\ \sin(\Theta) \\ \cos(\Theta) \end{pmatrix} \qquad \hat{e}_{p-in} = \begin{pmatrix} \cos(\gamma) \\ \sin(\gamma) \\ 0 \end{pmatrix} \qquad \hat{e}_{p-out} = \begin{pmatrix} \cos(\Pi) \\ \sin(\Pi)\cos(\Theta) \\ \sin(\Pi)\sin(\Theta) \end{pmatrix} \qquad \text{(1-a,b,c)}$$

The scattered wave vector $\overrightarrow{\Delta k}$ is defined as follow: $\overrightarrow{\Delta k} = 2\vec{k} - \vec{K} = \begin{pmatrix} 0 \\ -K\sin(\Theta) \\ 2k - K\cos(\Theta) \end{pmatrix}$ (2)

# 2)Expression of the SHS Intensity

## 2.1) Incoherent Second Harmonic Scattering

The Second Harmonic Scattering of uncorrelated molecule in solution is usually referred as Hyper Rayleigh Scattering (HRS) and can be expressed as:

$$I_{HRS}\left(\hat{e}_{p-in}, \hat{e}_{p-out}, \hat{e}_n\right) = I_{HRS}(\gamma, \Pi, \Theta) = \langle \vec{\beta}_{eff}(\gamma, \Pi, \Theta).\vec{\beta}^{*}_{eff}(\gamma, \Pi, \Theta)\rangle \qquad (3)$$

Here $\vec{\beta}_{eff} = (\hat{e}_n \times \overleftrightarrow{\beta}: \hat{e}_p\hat{e}_p \times \hat{e}_n)$ and $\overleftrightarrow{\beta}$ is the second order hyperpolarisability of the molecule expressed in the laboratory frame. The link between the expression of $\overleftrightarrow{\beta}$ in the laboratory and microscopic frames is:

$$\beta_{IJK,Labo}(\varphi, \theta, \psi) = \sum_i \sum_j \sum_k T_{Ii}(\varphi, \theta, \psi) T_{Jj}(\varphi, \theta, \psi) T_{Kk}(\varphi, \theta, \psi)\, \beta_{ijk,microscopic} \qquad (4)$$

The angle $\varphi, \theta, \psi$ are defined in the figure 2 and the transformation matrix $T(\varphi, \theta, \psi)$ between the microscopic and the laboratory frame is given within the 'ZYZ' convention:

$$T(\varphi, \theta, \psi) =$$

$$\begin{pmatrix} \mathrm{Cos}[\theta]\,\mathrm{Cos}[\varphi]\,\mathrm{Cos}[\psi] - \mathrm{Sin}[\varphi]\,\mathrm{Sin}[\psi] & -\mathrm{Cos}[\psi]\,\mathrm{Sin}[\varphi] - \mathrm{Cos}[\theta]\,\mathrm{Cos}[\varphi]\,\mathrm{Sin}[\psi] & \mathrm{Cos}[\varphi]\,\mathrm{Sin}[\theta] \\ \mathrm{Cos}[\theta]\,\mathrm{Cos}[\psi]\,\mathrm{Sin}[\varphi] + \mathrm{Cos}[\varphi]\,\mathrm{Sin}[\psi] & \mathrm{Cos}[\varphi]\,\mathrm{Cos}[\psi] - \mathrm{Cos}[\theta]\,\mathrm{Sin}[\varphi]\,\mathrm{Sin}[\psi] & \mathrm{Sin}[\theta]\,\mathrm{Sin}[\varphi] \\ -\mathrm{Cos}[\psi]\,\mathrm{Sin}[\theta] & \mathrm{Sin}[\theta]\,\mathrm{Sin}[\psi] & \mathrm{Cos}[\theta] \end{pmatrix} \qquad (5)$$

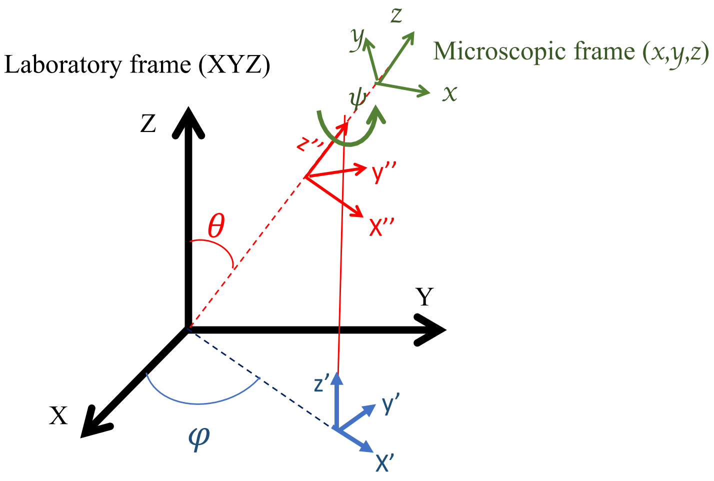


Figure 2: Definition of the angles $\varphi, \theta, \psi$ which describe the orientation of the scattered object in the laboratory frame.

Finally, the bracket means the average over all the orientation of the scattered object.

$$\langle\,.\,\rangle = \int_{\theta=0}^{\pi}\int_{\varphi=0}^{2\pi}\int_{\psi=0}^{2\pi}.\, sin\theta d\theta d\varphi d\psi \qquad (6)$$

The development of the above equations gives the following expression of $\vec{\beta}_{eff}$ expressed in term of the tensor $\beta$ components.

$$\vec{\beta}_{eff}(\gamma,\Theta) =$$
$$\begin{pmatrix} \beta_{XXX}.cos^2(\gamma) + 2\,\beta_{XXY}.\cos(\gamma)\sin(\gamma) + \beta_{XYY}.sin^2(\gamma) \\ [\beta_{YYY}.cos^2(\gamma) + 2\,\beta_{YXY}.\cos(\gamma)\sin(\gamma) + \beta_{YYY}.sin^2(\gamma)].cos^2(\Theta) - [\beta_{ZYY}.cos^2(\gamma) + 2\,\beta_{ZXY}.\cos(\gamma)\sin(\gamma) + \beta_{ZYY}.sin^2(\gamma)].\cos(\Theta)\sin(\Theta) \\ -[\beta_{YYY}.cos^2(\gamma) + 2\,\beta_{YXY}.\cos(\gamma)\sin(\gamma) + \beta_{YYY}.sin^2(\gamma)].\cos(\Theta)\sin(\Theta) + [\beta_{ZYY}.cos^2(\gamma) + 2\,\beta_{ZXY}.\cos(\gamma)\sin(\gamma) + \beta_{ZYY}.sin^2(\gamma)]\,sin^2(\Theta) \end{pmatrix} (7)$$

## 2.2) Coherent Second Harmonic Scattering of N fully correlated molecules

The Second Harmonic Scattering of N fully correlated molecule in solution can be expressed as:

$$I_{SHS}(\hat{e}_{p-in}, \hat{e}_{p-out}, \hat{e}_n) = I_{SHS}(\gamma, \Pi, \Theta) = \langle \vec{\beta}_{eff}(\gamma, \Pi, \Theta).\vec{\beta}^*_{eff}(\gamma, \Pi, \Theta)\rangle \qquad (8)$$

$$\text{Here } \vec{\beta}_{eff} = (\hat{e}_n \times \overleftrightarrow{\beta_t}: \hat{e}_p\hat{e}_p \times \hat{e}_n) \quad \text{with } \overleftrightarrow{\beta_t} = \sum_j^N \overleftrightarrow{\beta}_j e^{i\overrightarrow{\Delta k}.\vec{r}_j} \qquad (9\text{-}10)$$

where $\overleftrightarrow{\beta}_j$ is the second order hyperpolarisability of the molecule j located at the position $\vec{r}_j$ and $\overrightarrow{\Delta k}$ has been defined above. To perform the I<sub>SHS</sub> calculation, the user needs to specify the position $\vec{r}_j = (X'_j, Y'_j, Z'_j)$ and orientation $(\varphi'_j, \theta'_j, \psi'_j)$ of each molecule in the mesoscopic frame.

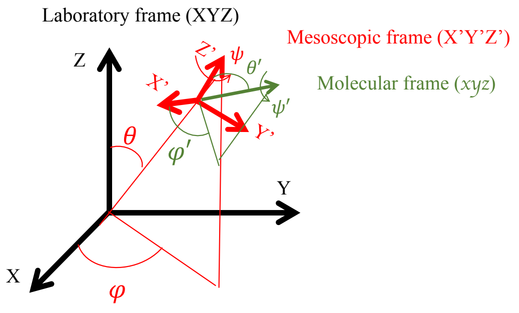


Figure 3: Definition of the angles $\varphi$, $\theta$, $\psi$ and $\varphi'$, $\theta'$, $\psi$' which describe the orientation of the scattered object in the laboratory and mesoscopic frame.

## 2.3) Second Harmonic Scattering by a sphere or a cylinder

In the case of colloidal object in solution like spheres or cylinder , we can expressed the $I_{SHS}$ intensity by introducing an effective nonlinear susceptibility $\overleftrightarrow{\Gamma^{(2)}}$ , which represents the nonlinear response of the entire particle surface[1] :

$$I_{SHS}\left(\hat{e}_{p-in},\hat{e}_{p-out},\hat{e}_n\right) = I_{SHS}(\gamma,\Pi,\Theta) = \langle \vec{\beta}_{eff}(\gamma,\Pi,\Theta).\vec{\beta}^{*}_{eff}(\gamma,\Pi,\Theta)\rangle \tag{11}$$

Here $\vec{\beta}_{eff} = (\hat{e}_n \times \overleftrightarrow{\Gamma^{(2)}}: \hat{e}_p\hat{e}_p \times \hat{e}_n)$ (12)

and $\overleftrightarrow{\Gamma^{(2)}} = \oiint_{surface\ of\ the\ particle} \overleftrightarrow{\chi^{(2)}} \,.\, e^{i\overrightarrow{\Delta k}.\vec{r}} d\vec{r}$ (13)

where $\overleftrightarrow{\chi^{(2)}}$ is the surface second order nonlinear susceptibility of the particle.

[1] Roke and al Phys. Rev. B **70**, 115106 (2004)

Chapter 3

# Incoherent Second Harmonic Scattering : HRS program

## 1) The different input/output

The program HRS compute the polarization resolved incoherent SHS Intensity at any specified scattering angle. The only input parameters of this program is:

- the microscopic hyperpolarizability tensor. Those components are passed through the input_beta file.

  Format of the input_beta file:
  line 1: comments
  line 2:"xxx" space value
  line3: "xxy" space value

And so on…

With the keyword "polarplot_single", the output calculation are the coefficients $a_{\Pi}^{\Theta},\ b_{\Pi}^{\Theta},\ c_{\Pi}^{\Theta}$ where $\Pi$ =V or H and $\Theta$ is the scattering angle. They are defined by:

$$I_{SHS}{}^{\Theta}(\gamma,\Pi) = a_{\Pi}^{\Theta}.cos^4(\gamma) + b_{\Pi}^{\Theta}.cos^2(\gamma)sin^2(\gamma) + c_{\Pi}^{\Theta}.sin^4(\gamma) \tag{14}$$

The comparison with equation (7) gives:

$$a_{\mathrm{V}}^{\Theta} = \langle \beta_{XXX}\beta_{XXX}^{*} \rangle \tag{15}$$

$$c_{\mathrm{V}}^{\Theta} = \langle \beta_{XYY}\beta_{XYY}^{*} \rangle \tag{16}$$

$$b_{\mathrm{v}}^{\Theta} = \langle 4\beta_{XXY}\beta_{XXY}^{*} + \beta_{XXX}\beta_{XYY}^{*} + \beta_{XXX}^{*}\beta_{XYY} \rangle \tag{17}$$

$$a_{\mathrm{H}}^{\Theta} = \langle \beta_{YXX}\beta_{YXX}^{*}cos^2(\Theta) + \beta_{ZXX}\beta_{ZXX}^{*}sin^2(\Theta) - \cos\Theta\,\sin\Theta(\beta_{YXX}\beta_{ZXX}^{*} + \beta_{ZXX}\beta_{YXX}^{*}) \rangle \tag{18}$$

$$c_{\mathrm{H}}^{\Theta} = \langle \beta_{YYY}\beta_{YYY}^{*}cos^2(\Theta) + \beta_{ZYY}\beta_{ZYY}^{*}sin^2(\Theta) - \cos\Theta\,\sin\Theta(\beta_{YYY}\beta_{ZYY}^{*} + \beta_{ZYY}\beta_{YYY}^{*}) \rangle \tag{19}$$

$$\begin{aligned} b_{\mathrm{H}}^{\Theta} = \langle & cos^2(\Theta)[4\beta_{YXY}\beta_{YXY}^{*} + \beta_{YXX}\beta_{YYY}^{*} + \beta_{YXX}^{*}\beta_{YYY}] + sin^2(\Theta)[4\beta_{ZXY}\beta_{ZXY}^{*} + \beta_{ZXX}\beta_{ZYY}^{*} + \\ & \beta_{ZXX}^{*}\beta_{ZYY}] - \cos\Theta\,\sin\Theta[4\beta_{YXY}\beta_{ZXY}^{*} + 4\beta_{ZXY}\beta_{YXY}^{*} + \beta_{YXX}\beta_{ZYY}^{*} + \beta_{YXX}^{*}\beta_{ZYY} + \beta_{ZXX}\beta_{YYY}^{*} + \\ & \beta_{ZXX}^{*}\beta_{YYY}] \quad ) \rangle \end{aligned} \tag{20}$$

Additionnal coefficient I$_2$ and I$_4$ are also given. They are defined by:

$$I_2^{\Theta,\Pi} = \frac{4(a_{\Pi}^{\Theta} - c_{\Pi}^{\Theta})}{(3a_{\Pi}^{\Theta} + b_{\Pi}^{\Theta} + 3c_{\Pi}^{\Theta})} \tag{21}$$

$$I_4^{\Theta,\Pi} = \frac{(a_{\Pi}^{\Theta} - b_{\Pi}^{\Theta} + c_{\Pi}^{\Theta})}{(3a_{\Pi}^{\Theta} + b_{\Pi}^{\Theta} + 3c_{\Pi}^{\Theta})} \tag{22}$$

The program will also show a graphic with the polarization plot.

With the keyword “polarplot_integrate”, the program computes the polarization plot for a collection of different scattering angle between $\Theta_1$ and $\Theta_2$ as depicted on the scheme:

Finally, the program can also compute the angle resolved incoherent SHS Intensity for different polarization states:

- $\mathrm{IH}(0°) = I_{HRS}(\gamma = 0°, \Pi = 90°, \Theta)$
- $\mathrm{IH}(90°) = I_{HRS}(\gamma = 90°, \Pi = 90°, \Theta)$
- $\mathrm{IV}(0°) = I_{HRS}(\gamma = 0°, \Pi = 0°, \Theta)$
- $\mathrm{IV}(90°) = I_{HRS}(\gamma = 90°, \Pi = 0°, \Theta)$

The user has to specify the number of point between 0 and 180° that he want to compute.

# 2) Command line

In a terminal and in the PySHS_V2 directory, enter the following command:

```
>./Work/HRS <Keyword>  <inputbeta>  <outputfile>
```

where <Keyword> is equal to “polarplot_single” or “polarplot_integrate” or “angle_scattering” depending on the computation you want to perform. inputbeta, and outputfile are respectively the name of the input file containing the hyperpolarizability of the molecule, and the output file.

To plot the polarization plot with gnuplot, (http://www.gnuplot.info), open gnuplot and enter in the command:

```
>plot for [col=2:3] "out_plot" using 1:col with lines title columnheader
```

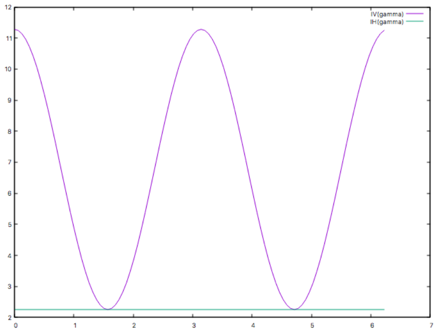


Figure 4: Polarization plot using gnuplot

Chapter 4

# Coherent Second Harmonic Scattering of Supramolecular structures : SHS Module

## 1) The different input/output

The program SHS compute the polarization resolved SHS Intensity at an angle of respectively Θ = 90° or Θ = 0° or can also compute the angle resolved SHS Intensity for different polarization state as explained in chap 3. The user has to specify the number of point to be compute.

Different input parameters have to be given:

- microscopic hyperpolarizability tensor components are expressed in a file input_beta like in chap 3.
- the position and orientation of each dipole of the aggregate are expressed in the mesoscopic frame in an input_file2. The format of this input file is a 6 columns per line containing in the order$(\varphi'_j, \theta'_j\ , \psi'_j, x'_j, y'_j, z'_j)$ which represent the orientation and position of the molecule j. The number of line is the number of molecule in the aggregate. The angles $\varphi'_j, \theta'_j\ , \psi'_j$ **have to be given in radian.**

Some specific script also given in ./Scr folder can generate these input files for different geometry (sphere, cylinder …)

- the refractive index, the laser wavelength (expressed in nanometer), the number of dipole in the aggregate. All these argument are asked by the program.

The output calculation are the coefficient $a_\Pi^\Theta,\ b_\Pi^\Theta,\ c_\Pi^\Theta$ , $I_2^{\Theta,\Pi}$ and $I_4^{\Theta,\Pi}$ as explained in the HRS module. Additionnal coefficient $b1_\Pi^\Theta$ and $b2_\Pi^\Theta$ are also given. Indeed, in the developpment of equation (3) appears 2 terms that are zero in HRS calculus but may be eventually no vanished in some special case in SHS calculus, so their values are also given in the output. Those term are defined by:

$$I_{SHS}{}^{\Theta}(\gamma,\Pi) = a_\Pi^\Theta.cos^4(\gamma) + b_\Pi^\Theta.cos^2(\gamma)sin^2(\gamma) + c_\Pi^\Theta.sin^4(\gamma) + b1_\Pi^\Theta cos^3(\gamma)sin\ \ (\gamma) + b2_\Pi^\Theta cos\ \ (\gamma)sin^3(\gamma) \quad (26)$$

# 2)Command line

In a terminal, enter the following command:

```
>./Work/SRS <Keyword>  <inputbeta>  <input_position_orientation> <outputfile>
```

where <Keyword> is equal to polarplot_single or polarplot_integrate or angle_scattering depending on the computation you want to perform. inputbeta, input_orientation and outputfile are respectively the name of the input file containing the hyperpolarizability of the molecule, the name of the input file containing the orientation and position of the molecules and the output file.

To plot the polarization plot with gnuplot, open gnuplot

and enter in the command:

```
>plot for [col=2:3] "out_plot" using 1:col with lines title columnheader
```

To see the position of each dipole in space, open gnuplot and enter in the command:

```
>splot "out_arrows_gnuplot" with vectors filled
```

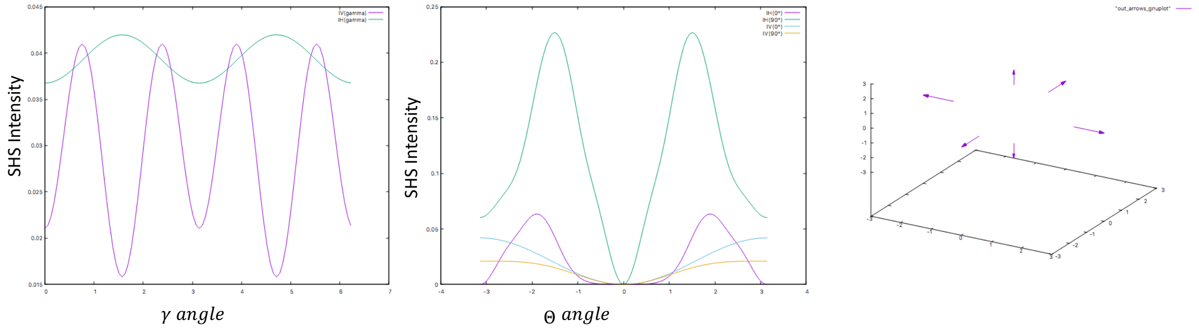


Figure 5: polarization plot at 90° (left), angle resolved SHS Intensity (middle) and position orientation of each dipole in the supramolecular aggregate (right).

Chapter 5

# Coherent Second Harmonic Scattering of Colloidal object : Sphere Module

## 1) The different input/output

The programs sphere_SHS and cylinder_SHS compute the polarization resolved SHS Intensity of a sphere or a cylinder with a surface nonlinear susceptibility at an angle of respectively $\Theta = 90°$ and $\Theta = 0°$ or can also compute the angle resolved SHS Intensity for different polarization state as explained in chap 3.

Different input parameters have to be given:

- Surface second order susceptibility tensor components are expressed in a file input_ki(2) like in chap 3.
- the refractive index, the laser wavelength (expressed in nm), the radius of the sphere. All these argument are asked by the program.

- The output calculation are the coefficient $a_{\Pi}^{\Theta}$, $b_{\Pi}^{\Theta}$, $c_{\Pi}^{\Theta}$ , $I_{2}^{\Theta,\Pi}$ and $I_{4}^{\Theta,\Pi}$ for the polarplot calculation and the $\mathrm{IH}(0°)$; $\mathrm{IH}(90°)$; $\mathrm{IV}(0°), \mathrm{IV}(90°)$ for the angular distribution calculation.

The programs cylinder_shell_SHS (not available on this version) compute the polarization resolved SHS Intensity of a cylinder (see Figure 6) of radius R and length d with a shell nonlinear susceptibility extension of length l at an angle of respectively $\Theta = 90°$ and $\Theta = 0°$ or can also compute the angle resolved SHS Intensity for different polarization state as explained in chap 3.

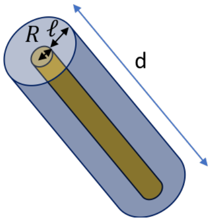


Figure 6 : the geometry of the shell cylinder

Chapter 6

# Additional Scripts and Perspectives

## 1) Scripts to generate orientation-position inputfile in SHS module

- “script_sphere.py” : It generates the position of N dipoles onto a sphere of specified radius and with a radial orientation. This script is based on this code[2]. The input parameters asked by the program are the radius of the sphere and number of dipoles.

- “script_sphere _random.py”: It generates the position of N dipoles onto a sphere of specified radius and with a specified normal distribution orientation around the radial direction. The input parameters asked by the program are the radius of the sphere, number of dipoles and the standard deviation (sigma) of the normal distribution around the radial direction.

- “script_spheroid.py”: It generates the position of N dipoles onto a spheroid of specified radius a and c and with a normal orientation.

- “script_spheroid_random.py”: It generates the position of N dipoles onto a spheroid of specified a and c radius and with a specified normal distribution orientation around the normal direction.

- “script_cylinder.py”: It generates the position of dipoles uniformally spaced onto a cylinder of specified length and radius with a normal orientation.

- “script_cube.py”: It generates the position of dipoles uniformally spaced onto a cube of specified length with a normal orientation.

[2] https://bduvenhage.me/geometry/2019/07/31/generating-equidistant-vectors.html
https://www.openprocessing.org/sketch/41142

- “script_cylinder.py”: It generates the position of N dipoles onto a cylinder of specified length and radius and with a normal orientation.

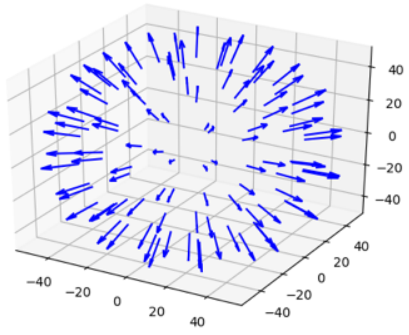
- “script_cylinder_random.py”: It generates the position of N dipoles onto a cylinder of specified length and radius and with a specified normal distribution orientation around the normal direction.

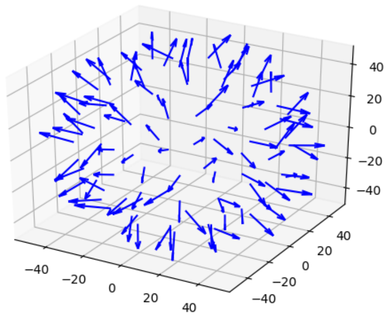
- “script_cylinder_shell.py”: It generates the position of N dipoles onto a shell surrounding a cylinder of specified extension l1, length l2 and radius a and with a specified normal distribution orientation around the normal direction.

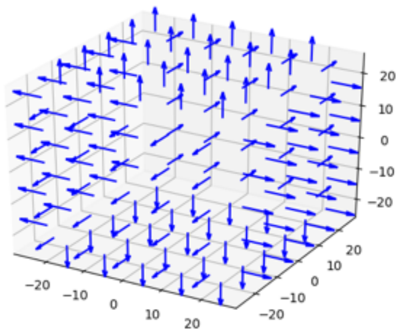
In all script, the command line is:

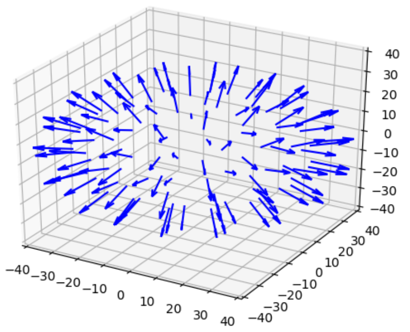
```
>python3  Src/script_XXX.py Work/output_script
```

Where “XXX” is “sphere” or “sphere_random” or ”spheroid” or “cylinder” or “cube”.









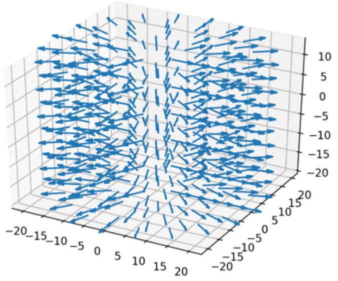


Figure 7 : Some dipoles organization obtained with the various script, in the order: script_sphere.py (radius =50, N=100), script_sphere_random.py (radius =50, N=100, sigma=0.5), script_cube.py (length=50, N=100), script_spheroid.py (a=40,c=20, N=100) and script_cylinder_shell.py