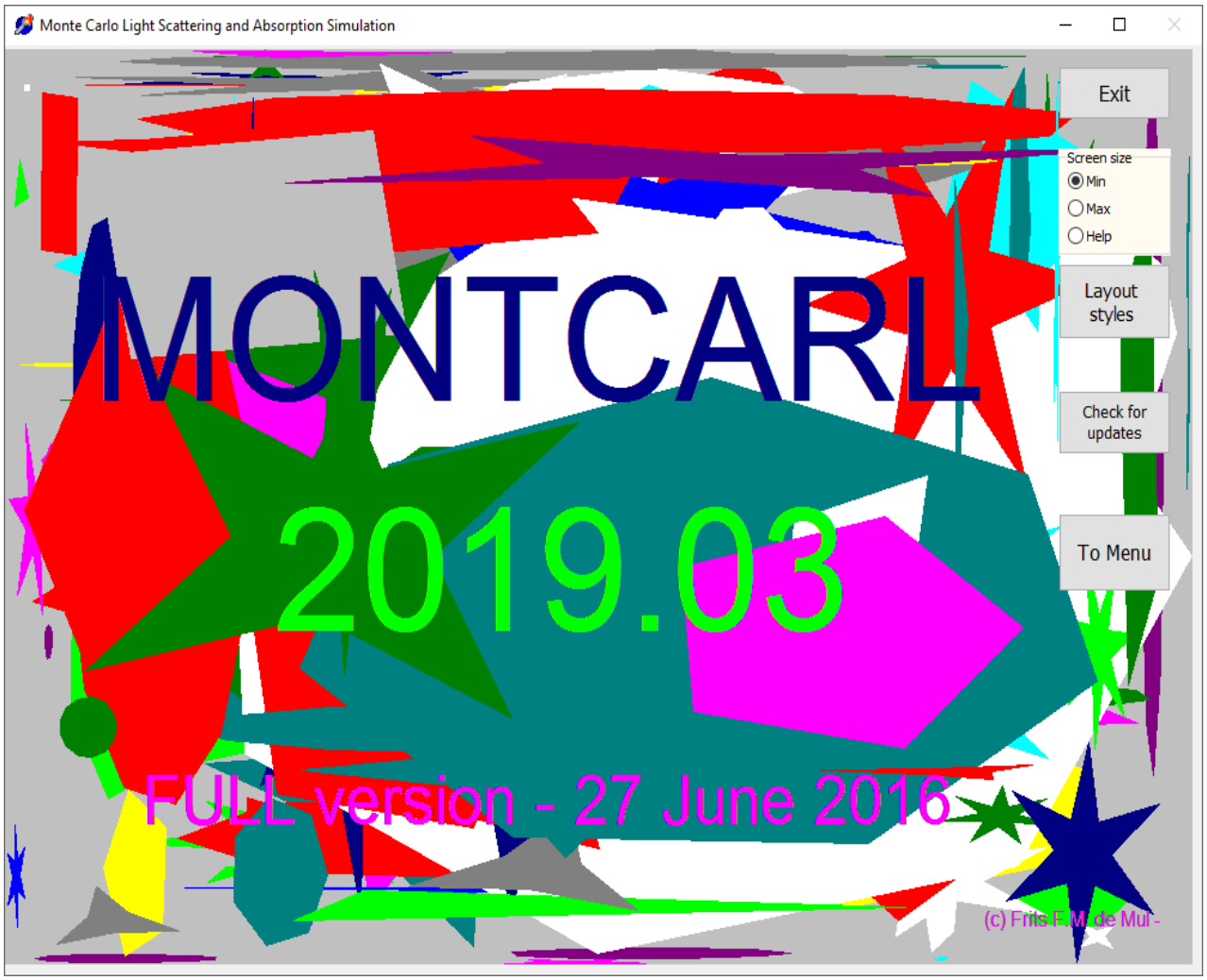


MONTCARL

MONTE-CARLO

SIMULATION

MONTCARL

* *

M O N T E - C A R L O S I M U L A T I O N
O F L I G H T S C A T T E R I N G A N D A B S O R P T I O N ,
F L U O R E S C E N C E A N D R A M A N - S C A T T E R I N G
I N T U R B I D M E D I A

Frits F.M. de Mul – www.demul.net/frits - June 2016

* *

- F.F.M. de Mul, M.H. Koelink, M.L. Kok, P.J. Harmsma, J. Greve, R. Graaff, J.G. Aarnoudse, "Laser Doppler Velocimetry and Monte Carlo Simulations on Models for Blood Perfusion in Tissue", Applied Optics, vol. 34, 1995, p. 6595-6611.
- De Mul, F.F.M.; Monte-Carlo simulation of Light transport in Turbid Media, Chapter 12 in: Handbook of Coherent Domain Optical Methods, Biomedical Diagnostics, Environment and Material Science, Tuchin, Valery V. (Ed.), 2004, XLII, 1004 p. (2-volume-set), Kluwer Publishers, Hardcover ISBN: 1-4020-7576-6, pages 465-533.
- De Mul, F.F.M.; Monte-Carlo simulation of Light transport in Turbid Media, Chapter 15 in: Handbook of Coherent Domain Optical Methods, Biomedical Diagnostics, Environment and Material Science, Tuchin, Valery V. (Ed.), 2013, Springer Publishers, Hardcover ISBN: 978-1-4614-5176-1, pages 593-661.
-

==== 0. GENERAL OBJECTIVE OF THE PROGRAM ===

The program calculates Monte-Carlo simulations of LIGHT SCATTERING and/or ABSORPTION (optional: followed by FLUORESCENCE or RAMAN-scattering, or PHOTOACOUSTICS) in turbid media, like tissue.

The SAMPLE may consist of a single or more LAYERS, each with its own absorption and scattering data, in the form of concentrations of scattering particles embedded in a medium.
In order to register DOPPLER spectra, to each type of scattering particle a certain velocity vector can be given.
In each layer a number of separately defined structures (called "OBJECTS") with rectangular, cylindrical, spherical, torusses or conical shape) may be present, with similar characteristics as the layers.
With those objects e.g. blood vessels can be mimicked.
Also an oblique mirror plane can be inserted.
Furthermore, the layers may be subdivided into sublayers (depth pixels, see below).

.

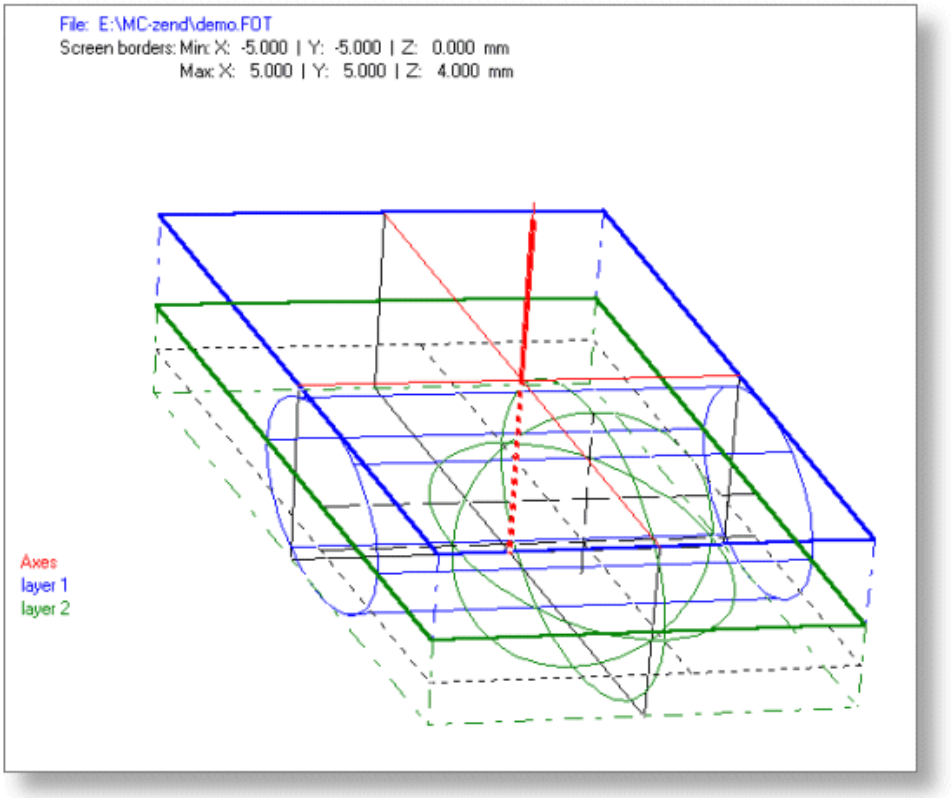


Fig. 1. Structure plot of a two-layer system with a horizontal cylindrical tube and asphere (see section 2), filled with various concentrations of scattering/absorbing particles. Laser light (here diverging beam) injected around Z-axis.
NB. All plots in this document were produced using the MONTCARL-program output facilities.
-→ MORE FIGURES AT THE END OF THIS REPORT.

There are a number of options: internal light sources and ecternal beams, internal and external detection, plots of distributions of various variables (photon positions, path length distributions, photon tracking, layer+object structure plots, frequency modulation by Fourier transformations, angular distributions, photoacoustic response plots,.etc. etc.

==== 1. PHOTON TRANSPORT : LAYERS AND OBJECTS ====

The program calculates Monte-Carlo simulations of LIGHT SCATTERING and/or ABSORPTION (optional: followed by FLUORESCENCE or RAMAN-scattering, or PHOTOACOUSTICS) in turbid media, like tissue.

The SAMPLE may consist of a single or more LAYERS, each with its own absorption and scattering data, in the form of concentrations of scattering particles embedded in a medium.
In order to register DOPPLER spectra, to each type of scattering particle a certain velocity vector can be given.
In each layer a number of separately defined structures (called "OBJECTS"): with rectangular, cylindrical, spherical or conical shape) may be present, with similar characteristics as the layers.
With those objects e.g. blood vessels can be mimicked.

Also an oblique mirror plane can be inserted.
Furthermore, the layers may be subdivided into sublayers (depth pixels, see below).

==== 2. MONTCARL - README.1ST - file ====

The Montcarl package contains following files:
- MC20xx.EXE : executable file (Windows-version),(program version 20.xx)
- README.1st : this introduction
- MONTCARL.TXT : this ASCII-text file with HELP information.
- DEMO.INP : This file is a demo-input file with some structures in the scattering medium.
- DEMO.MIE : A file with the scattering function for DEMO.INP

==== 3. INPUT AND OUTPUT OF PHOTON DATA ====

= INPUT FILES:
- *.INP FILE, *.LST AND *.MIE-FILE(S): data with settings for simulations. For details: see below.
- *.RSP : for type-R objects: coordinates of random spheres.

= OUTPUT FILES: The program has the following file output:
- *.FOx : (x=0..9,T) containing the photon data, as described below;
- *.INF : Ascii-file containing general data about the simulation, such as:
  + total number of photons reflected, absorbed, transmitted;
  + average numbers;
  + data of the scatterers, layers and objects;
- *.2D : 2-dim. plots of the variables of ALL detected photons;
- *.3Da : 3-dim. plots of the <scatter depth> or absorption depth of ALL detected photons.
- *.3Dd : id. of Doppler photons.
- *.PAT : file of tracking path xyz-positions (optional) (in units as with storage in *.FOT-file, see below), each path preceded by a dummy position: x,y,z = 9999,9999,# of following positions)
- *.DIK : used by the program: internal values.
- *.MCn : (n=1,2..) temporary files with settings;

In case a simulation run is interrupted during execution:
- .IRP : data (in internal format) to resume the simulation.

In case calculation of photo-acoustic response is available:
- *.ABS : 3-dim. voxels with # of absorbed photons.
- *.PAI : settings to calculate photo-acoustic time response
- *.PAC : photo-acoustic time-response results at array of detectors.
- *.PAG : id. for grouped array detector.

For the formats used, see below.

BEFORE RUNNING THE PROGRAM:

1. Create directories (= folders) for:
   *.MIE-files: Scattering functions (and copy DEMO.MIE in it);
   *.INP-files: Input-files (ASCII), and *.LST-files (Ascii: series list);
   *.FOT-files (and other output files): Storage of detected photons.
2. Copy *.INP- and *.MIE-files in the respective directories.
3. Start the program, read DEMO.INP and change its data lines containing the MIE- and FOT-directory names, using the EDIT-INP option, or with a text processor.

==== 4. PHOTOACOUSTICS AND FLUORESCENCE/RAMAN ====

The program is also capable of calculating the FLUORESCENCE/RAMAN or
PHOTOACOUSTIC response of photons absorbed inside the structure,
to be detected at an array of detectors at the surface.

The calculation consists of several steps:
  1. Calculate of a FOT-file with positions of photons, absorbed at
     the wavelength of absorption;
     and a PAT-file with tracking path positions (optional);
  2. (FLUORESCENCE/RAMAN): Using this file as photon source, calculate
     scattering at the wavelength of emission with a fresh INP-file.
     To handle a fluorescence/Raman SPECTRUM, a series of new INP-files
     can be created and simulated, with varying laser wavelengths.
  3. (PHOTOACOUSTICS): Using this file as source of acoustic sources,
     calculate photo-acoustic response
     (can be done directly after calculation of absorption FOT-file).
     For details: see option: "Photoacoustics: details"

==== 5. PHOTON INJECTION, TRANSPORT AND DETECTION ====

There are 3 options for entering photons:
1. from the outside world, using a beam;
2. from a single point in the inside of the layer system;
3. using absorbed photons from a previously made *.FOT-file.

Ad 1. A (laser) beam impinges onto the first layer (interface 0th-1st layer).
Ad 2. The direction and divergence of the photon distribution emerging from
      the source point is entered.
Ad 3. This is especially useful for calculating fluorescence or Raman. The
      photons start from their point of absorption, in isotropic direction.

FRESNEL RELATIONS are used for describing interface crossings.
Natural polarization has been assumed throughout.
The photon is followed when traveling through the sample.

Four modes are available: detection of SCATTERING by REFLECTION and/or
TRANSMISSION, or INTERNALLY, or of ABSORPTION,
optionally followed by FLUORESCENCE/RAMAN-scattering.

==== 6. DETECTION OF EMERGING PHOTONS ====

The program offers various ways of detecting emerging photons:
R: REFLECTION: upon refractive passage of the interface between two
                 layers N-1 and N (with N=1...nr.-of-layers; N=0:surface),
                 if directed towards negative Z-values (theta > 90 degr.)
S: REFLECTION: the same, but different storage.
T: TRANSMISSION: upon refractive passage of the backside of the layer system,
                 if directed towards positive Z-values (theta < 90 degr.)
I,J,K: INTERNAL: upon refractive passage of the surface of a sphere
                 or the curved surface of a cylinder tube or cone,
                 travelling from inside to outside.
A: ABSORPTION: upon absorption in a particle or a layer or an object
E: ALL: (= reflection + transmission + absorption)

In all cases the X and Y-position at the point of detection are stored.

In cases I,J,K and A, the real Z-position at that point is stored, but
with R, S and T, the avaraged scatter depth or the maximum depth is stored.
Case E is a mixture.

FLUORESCENCE/RAMAN: Detection of emerging photons as above.

==== 7. COORDINATE AXES, STORAGE UNITS AND ACCURACY ====

Z-AXIS perpendicular to the surface of the sample,
       pos. inward (Z>0 with increasing depth in the sample)
X and Y-axes: in plane of sample surface.

STORAGE UNITS:
= FOT-files created with progr.version < 2018:
   1 micron (default); or 10 micron (in following cases:
   (1) for X- and Y-coords: if detection field (see below) extends
       beyond +/- 32768 micron (= max. for "integers")
   (2) for Z-coords: if total thickness of layer system extends
       beyond 65535 micron (= max. for "words").
= FOT-files created with prog.version >=2018: no limitations.

When injected from above (Z<0) the photon starts at a depth of 0.1 um
underneath surface. The photon beam can be injected from within the
sample as well (use the "focus"-option).

UNITS for angles: 1 degree,   for Doppler frequency: 1 Hz.

==== 8. PROGRAM INTERRUPT, STOP AND BREAK ====

INTERRUPT: The execution of simulations (single or series) can be
  interrupted by pressing the Interrupt-button.
  The relevant data are stored in a file, named after the actual *.INP-file,
  or - in case of a series of simulations - after the
  first file in the series, and will have extension *.IRP.
  Afterwards the program will detect the presence of those interruption files
  and will ask the user about resumption.
  This *.IRP-file is also created or updated at each output-to-file instant,
  (i.e. after each 1000 photons) enabling resumption after a program crash.

STOP: Pressing the Stop-button will stop the program
  but no IRP-file will be created.

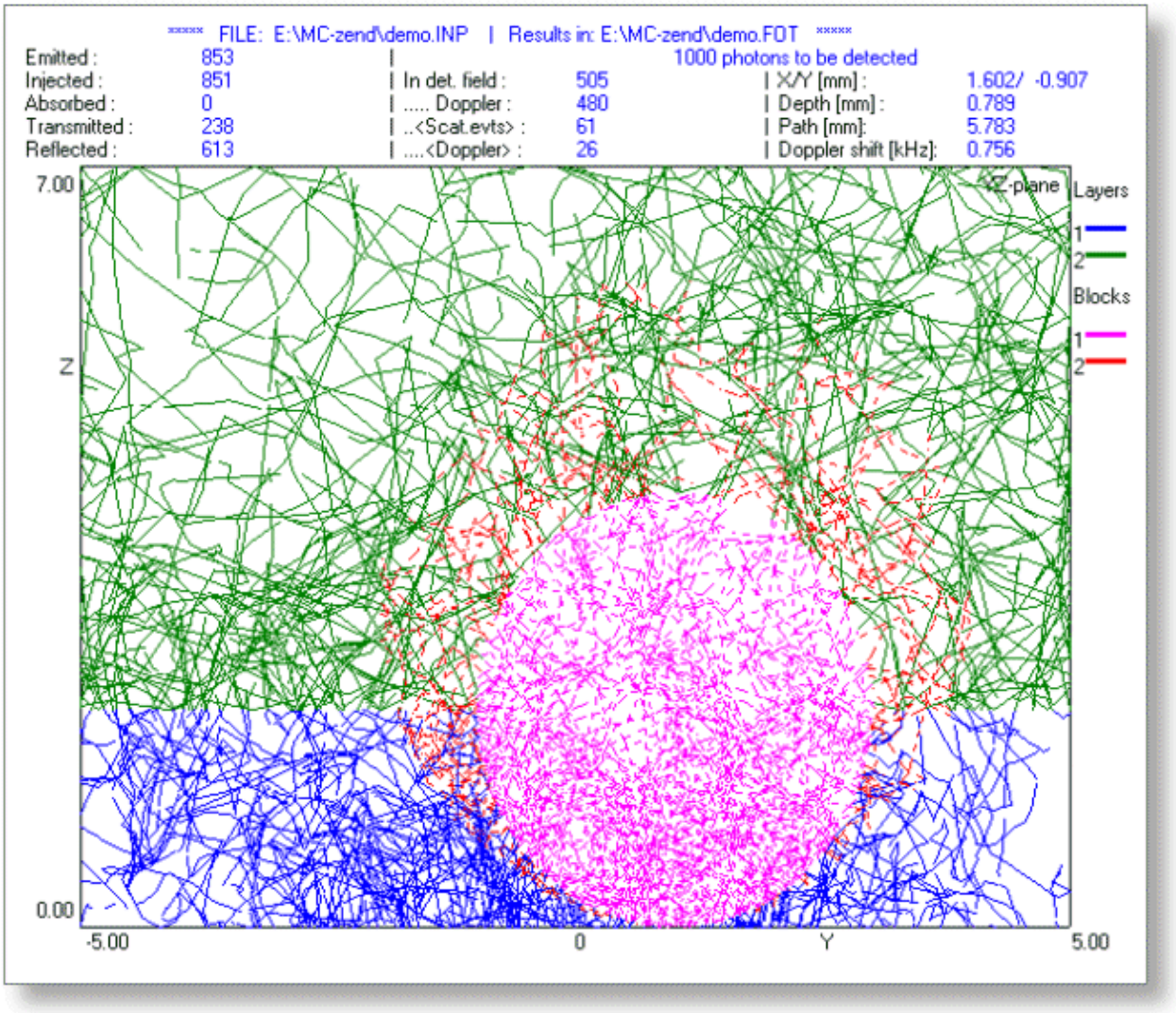


Fig. 2. Running graphics of the simulation process of the structure of fig.1. View in YZ-plane. Photons entering around pos (0,0,0). The tube (X-direction) and sphere can be seen.

==== 9. PHOTON SOURCE : wavelength, direction, dimensions, divergence ====

The program can handle :
- distributed sources (a FOT-file with absorbed photons)
- pencil beams (set divergences = 0);
- diverging beams;
- parallel beams;
- ring-shaped beams.

Each BEAM SHAPE: 2 options: Uniform/Gaussian; half-width HWHM to be entered.
If PENCIL beams and DIVERGING beams, the beam is assumed to be created in one point, called FOCUS. This focus point may be positioned within the layer system (Z>0) or in front of it (Z<0).
If PARALLEL BEAM, the convention is adopted that the Focus point always is situated at -1000 mm, i.e. in front of the layer system. When this value is inserted, the program assumes a lens at the surface, with a focal length of 1000 mm. In doing so a parallel beam will enter

the layer system. Here the divergence angles are used to determine
the width of the beam at the surface.
If RING-shaped beam, the Tilt angles theta and phi have a different
meaning (see below).

BEAM TILT: The mean direction of the beam can be tilted in 2 directions:
with respect to the Z-axis (details: see below) and within the XY-plane;
(1) TILT ANGLE OF LASER BEAM with respect to Z-axis (THETA-angle):
* Angle of beam direction vector with +Z-axis (+=inward) (0..180), or:
* Isotropic injection: two options: enter -1 or -2:
-1: according to: Distribution(t)=const. (t=theta)
then Cumul.funct.(t)=Integral[const.sin(t).dt] over 0..pi/2;
-2: according to: flux from isotropic radiance,
then Cumul.funct.(t)=Integral[const.cos(t).sin(t).dt] over 0..pi/2.
then direction will be randomized at point of injection.
(2) TILT ANGLE in XY-plane with respect to X-axis (PHI-angle):
if z-focus > 0 (within layer system) and source-in-focus: add 1000 to phi.

To enter beam spot size <>0: use (focus.z=-1000) & (divergence angles<>0)!

If RING-shaped beam:
Enter: MAX.ring radius + 1000 [mm] in stead of theta,
and MIN.ring radius [mm] in stead of phi. (Rectang. distr. only!)

DIAPHRAGM: can be used to limit the beam area at the surface (optional)
if wide open: enter 1000.

==== 10. OBJECTS : Rectangular, cylindrical, spherical, conical, mirror ====

The program can handle 11 types of "OBJECTS"
(in previous versions called "Blocks"):
- RECTANGLES: The sides of these objects are parallel to the main
coordinate axes (type B);
- CYLINDERS: 3 types: in X, Y, or Z-direction; (types X, Y and Z);
- SPHERES (type S), single or arranged in layers;
- RANDOM SPHERES (type R),
- OBLIQUE (type Q) ,
- CONICAL (type C),
- LENSES (type L), thick or thin,
- MIRROR PLANE (type M)
- (half) TORUS (type T).

Type-X and Y objects consist basically of a CYLINDER of INFINITE length,
parallel to the X or Y axis respectively.
The basic cylinder can be multiplied in the local Z=const plane,
by entering a repetition distance. This distance is in the
Y- or X-direction for the X- or Y-type respectively.
Type-Z objects consist basically of a CYLINDER of FINITE length,
parallel to the Z axis. The minimum and maximum number of the layers in
which the Z-object extends can be entered. This basic cylinder can be
multiplied in both X- and Y-direction, by entering a Repetition Distance.
Type-Q object is an OBLIQUE Z-object; no multiplication possible.
Type-S objects (SPHERES) can be multiplied in the same way as Z-objects.
Type-R objects (RANDOM SPHERES) are sub-objects,
situated in a cylindrical object (extending over 1 or more layers).
Position and radius are random; overlap allowed/excluded is optional.
Type-C objects (CONES) (in pos/neg X,Y or Z-dir.) are always single.

Type-M objects (NIRROR planes) are "thin" planes, with reflection
  determined by the refractive index. The direction of the normal vector
  is defined with directional theta and phi-angles.
Type L-objects (LENSES), thick or thin, transparent or turbid, with
  symmetry axis // Z-axis,
Type-T objects (half-TORUS) are half-torusses in XZ-plane,
  with symmetry axis // Y-axis, and the symmetry point on a layer boundary.
  From that boundary, the half-torus stretches out with the
  U-shape opening towards pos. or neg. Z-values
  (towards +Z-axis or -Z-axis respectively).
  NB. The symmetry point (axis point) of a half-torus MUST be
      on a layer boundary!

CONVENTION FOR INTERSECTING OBJECTS: inside overlapping regions:
  An object with lower number will penetrate in an object with higher number,
  and replace all properties of that object (and the layer where it is in).
  Example:
  If object 1 is situated within object 2, then the overlapping region will
  possess the properties of object 1.
  If object 2 is situated within object 1, then the overlapping region will
  possess the properties of object 1, with the effect that in that region
  object 2 seems not to exist, from the photon point of view.

==== 11. CALCULATION: SINGLE OR SERIES simulations (*.LST) ====

The program is made to handle a "single" simulation (conversationally),
or automatically a number of subsequent simulations (called a SERIES).
If a "SERIES" of simulations is wanted, a list of *.INP-files has to be
created, in the form of a *.LST-file (a text file,
with the same name as that of the first file in the list).
This creation can be done within the program, or using a word processor.
When "SINGLE", photon paths can be followed graphically.

==== 12. OVERVIEW OF FILES : input and output ====

NB. "*" stands for the file name (e.g. ABC00123.xxx ; xxx=extension)

= INPUT FILES: *.INP FILE, *.LST AND *.MIE-FILE(S): see below.

= OUTPUT FILES: The program has the following file output:
  - *.FOx : (x=0..9,T) containing the photon data, as described below;
  - *.INF : Ascii-file containing general data about the simulation, such as:
            + total number of photons reflected, absorbed, transmitted;
            + average numbers;
            + data of the scatterers, layers and objects;
  - *.2D : 2-dim. plots of the variables of ALL detected photons;
  - *.3Da : 3-dim. plots of the <scatter depth> or absorption depth
               of ALL detected photons.
  - *.3Dd : id. of Doppler photons.
  - *.PAT : file of tracking path xyz-positions (optional) (in units as with
            storage in *.FOT-file, see below), each path preceded by a
            dummy position: x,y,z = 9999,9999,# of following positions)
  - *.DIK : used by the program: internal values.
  - *.MCn : (n=1,2..) temporary files with settings;

  In case calculation of photo-acoustic response is available:
  - *.ABS : 3-dim. voxels with # of absorbed photons.

- *.PAI : settings to calculate photo-acoustic time response
- *.PAC : photo-acoustic time-response results at array of detectors.
- *.PAG : id. for grouped array detector.

==== 13. INPUT-FILES: *.MIE and *.INP: introduction ====

The program needs two (sets of) input-files:

1. A (set of) *.MIE-files (one for each type of scattering particle)
   (see below)

2. A *.INP-file: containing data about:
   - the laser beam;
   - the type of scattering particles and their scattering functions;
     (called: MIE-tables ; see below: MIE-files)
   - the layers and objects structure;
   - the detection window and the numerical aperture of the detector;
   - the type of calculation: reflection/transmission/absorption/all;
   - the nr. of photons to be injected or detected;
   - the depth pixel structure;
   - the directory for file output (*.FOT-files etc)

3. For R-type objects (random spheres):
   - *.RSP : coordinates of random spheres.

   A series of INP-files can be listed in a *.LST-file, which enables to
   perform a SERIES OF SIMULATIONS automatically.

==== 13a .INPUT FILE: *.MIE : Scattering or Phase Functions ====

The extension *.MIE is used as a general designation for scattering
functions, calculated using the Mie-formalism or otherwise. These
files have to be calculated before starting a simulation. This can be
done using the proper option at the start of the program.
There are several formalisms available:
-1) dipolar, -2) Rayleigh/Gans, -3) Mie,
-4) isotropic, -5) peaked forward (Gaussian profile),
-6) Henyey-Greenstein, -7) Gegenbauer, -8) cos^2.
Options 1)..3) calculate absolute values for the scattering
function (including SCATTERING CROSS SECTION values [micron^2]).
Options 4)..8) calculate phase functions only. These functions
are normalised and then artificially multiplied with the total scatt-
ering cross section obtained from a Mie-calculation.

To arrive at the proper SCATTERING COEFFICIENTS [mm^-1] of those types of
particles in the various layers and objects, in the *.INP-file the
CONCENTRATIONS [mm^-3] of these particles must be entered.
Multiplication of the concentration with the scattering cross section
results in the scattering coefficient [mm^-1] and the corresponding
mean free path [mm] in the layer or object. These values are used to
calculate the actual flight paths of the photons.

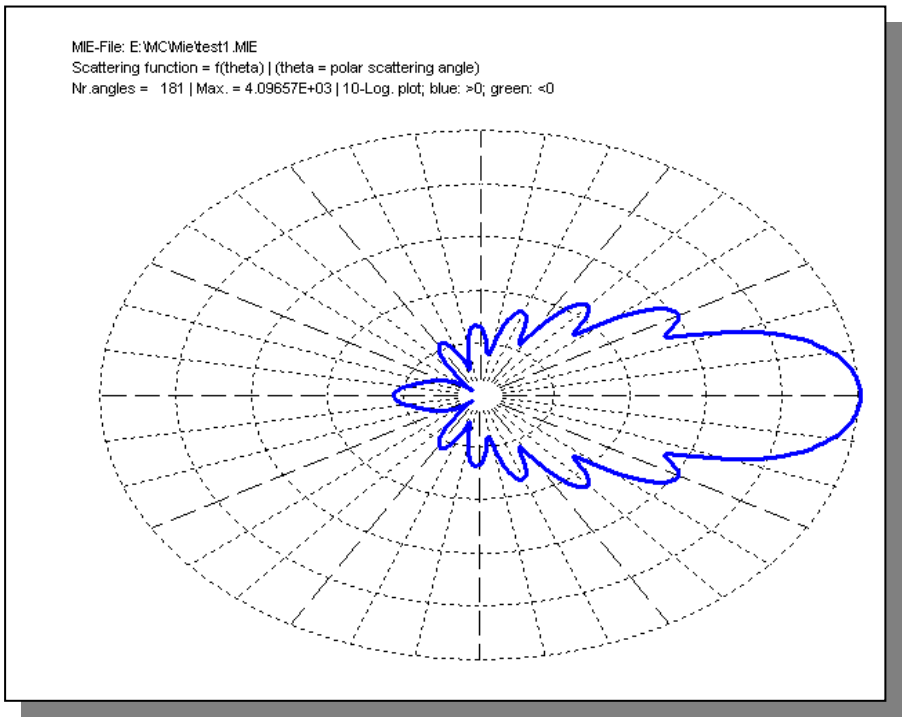


Fig. 3. Example of a MIE-file. Scattering function according to the Mie-formalism.

CONTENTS OF MIE-FILES: (file of extended: 10 bytes reals)
- nr. of angles, starting with theta=0 (zero), max. 181 values!
- for each angle:
  + angle (rads),
  + resulting scattering function value at this angle,
  + cumulative function value,
    (the cumul.function is normalised to 1 (one) at highest angle value)
- cumul.function at max. angle (before normalisation) * overall factor =
    = (aspect ratio)^2 * Q_scatt. /2 ,
    with Q_scatt = scattering cross section / geometrical cross section,
- relative refractive index of scattering particles with respect to medium,
- aspect ratio of the particles,
- for each contributing scattering function:
      sequence nr. and contributing factor (and g, if Henyey-Greenstein);
- wavelength in medium (optional)

Sequence nrs. of contributing functions in the program:

0. Overall multiplication factor	5. Henyey-Greenstein
1. Entered by keyboard or otherwise	6. Isotropic
2. Dipolar	7. Gegenbauer
3. Rayleigh-Gans	8. Peaked forward (Gaussian)
4. Mie	9. Cos^2

==== 13b. INPUT FILE: *.INP : input data about sample and scattering ====

For each simulation an Ascii-text file with input data is needed.
These files are called *.INP files. They may be created within the
  program itself, using the appropriate option in the start menu,
  or may be created using a word processor, in Ascii-type output files.
From one *.INP-file a SET of other *.INP files, in which several
  variables are altered subsequently, can be constructed.

For this purpose another option in the main menu is present.
This option is useful when planning a SERIES of subsequent simulations
  (*.INP-file names in the series can be listed in *.LST).

```
    20                      U                         1.300
** Photon source data         1.100                 0.000E+00
**                          0.000E+00                 -1.000
   780.0                       2.000                   1.000
  10.000                    ** Layer 2 **              3.000
  10.000                    1.000E-01                  3.000
   0.000                     0.000                     0.000
    0.00                       1.500                     1
  -10.00                    0.000E+00                 ** Calcul & Detect **
 100.000                       5.000                  S
  1.0000                    ** Object 1 **              -5.000
  1.0000                    X                           5.000
** Nr.                     1.000E-01                   -5.000
scatts/layers/objects       0.000                      5.000
**                             1.300                   1.000
     1                      0.000E+00                 ** Depth pixels **
     2                         1.000                     5
     2                         1.000                    25
E:\MC-zend                     2.000                  ** File output **
** Scatterer 1 **              2.000                  E:\MC-zend
demo                           0.000                  B
0.000E+00                        1                    L
** Layer 1 **               ** Object 2 **            ** Nr of photons **
1.000E-01                  S                          D1000
 1.000                     1.000E-01
X                           0.000
```

Fig.4. Example of an *.INP-file (file DEMO.INP, producing the structure of Fig.1.)

CONTENTS OF THE *.INP FILES

NB.=> Lines starting with "**" : Comment lines,
            will be skipped while reading.
            These lines may be inserted freely by the user.
   => Lines starting with ">" : data to be entered in *.INP-file.
   => All entries on separate lines (unless indicated otherwise)!!!!!

> ** PHOTON SOURCE (e.g. laser beam) **
> WAVELENGTH [nm]
> If photons to be produced from internal absorption (for fluoresc./Raman):
  Enter: Name (incl dir.+ ext.) of FOT-file containing absorbed photons.
  ==> JUMP to: input of "REFRACTIVE INDEX of medium in front of
                  the system of layers" !!!
> If photons produced from 1 single source point (e.g. laser beam or 1 point
  inside the layer system): enter:
  Beam DIVERGENCE angle (HWHM) : X-direction [degr.]
> Beam DIVERGENCE angle (HWHM) : Y-direction [degr.]
    (Gaussian/Rectangular distribution: enter angles x +1/-1)
  PARALLEL (//) beam:
    enter Depth of focus = -1000 mm
    (At focus=-1000 mm & HWHM=1 mm ==> angle = 0.001 rad = 0.057296 degr.)
    (to make // beam: program assumes a positive lens at the surface)

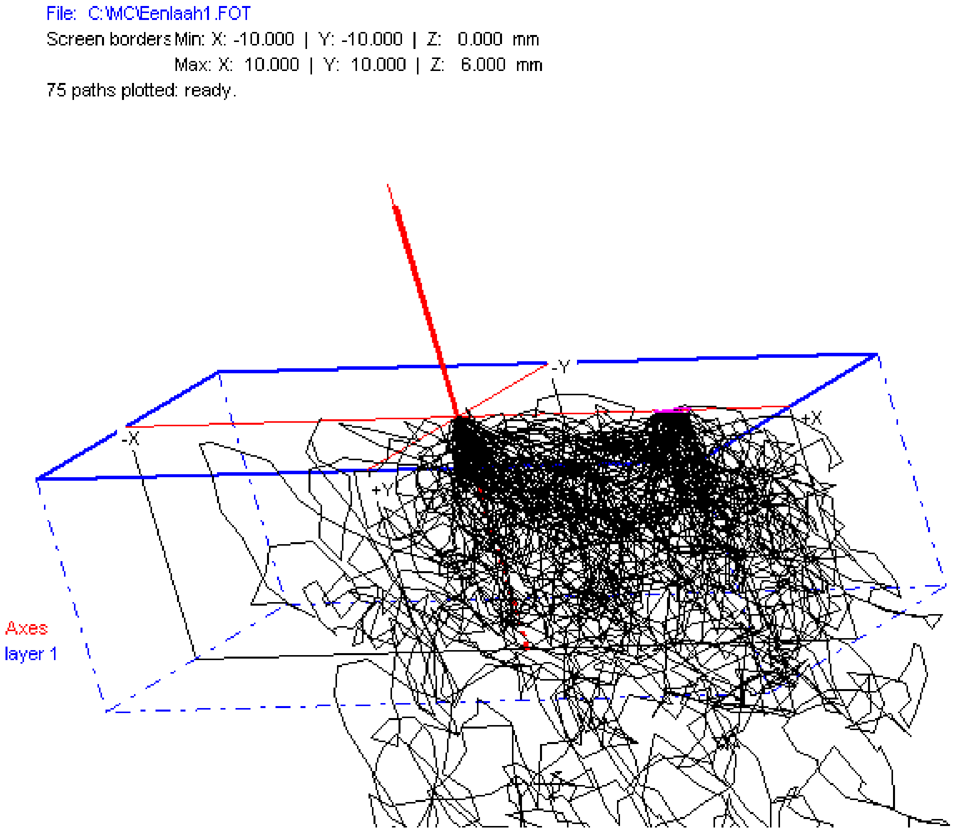
```
  RING-shaped beam: see below: "theta" and "phi".

> THETA: tilting angle of beam vector with +Z-axis: (0..180) [degr.]
  or: for Isotropic injection: enter -1, -2 or -3 (see at: LASER BEAM).
> PHI: angle of beam vector in XY-plane with +X-axis: (0..360) [degr.]
  (Default: source at surface;
   if source in focus at Z>0, thus INSIDE the layer system: add 1000 to phi)

  if RING-shaped beam:
    enter MAX.ring radius + 1000 [mm] in stead of theta,
    and   MIN.ring radius [mm] in stead of phi.
    (Rectang. distr. only!)

  FOCUS:
> Z-position of focus with respect to sample surface [mm] (-1000 for // beam)
  (Optional: X,Y-pos. of focus [mm]: at same line with Z-pos., ==>
    format: (e.g.) -3.32/1.245/-0.27 for Z/X/Y-pos. Watch the sequence !!)
  NB. If focus.Z=-1000, then // beam perpendicular to surface, independent of
      tilting angles!.
      If you like // beam, // tilting direction: take focus.Z=-999.99.

> DIAPHRAGM radius of beam at sample surface (*+1: circle; *-1: square) [mm]

> REFRACTIVE INDEX of the medium in front of the system of layers
> Id. of the medium behind the system of layers
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-

> ** NUMBER OF TYPES OF SCATTERERS, LAYERS AND objectS  **

> Number of Types of scatterers
     NB. Different types differ in RELATIVE Refr.index (w.r.t medium)
       and/or Aspect ratio (circumference/wavelength)
       and/or Mie-file (scattering function) !!

> Number of Layers

> Number of Objects

> Directory (folder) where the *.MIE-files are located.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-

FOR EACH TYPE OF SCATTERER :

>> ** SCATTERER # **               (#: nr.)
>> SCATTERING TABLE (*.MIE-file name, without extension .MIE)
    This table must have been created in advance of the simulations!
    For layers or objects without a predefined concentration or sigma_scatt
    [um^2] but with a known mu_scatt [mm-1], relat.refract.index and
    phase (or scattering) function of the particles, the PROCEDURE is:
    : create the proper Mie-file using the rel.refr.index by adjusting the
      aspect ratio at the proper wavelength;
    : insert this Mie-file in the *.INP-file and adjust the concentration
>> ABSORPTION CROSS SECTION (sigma_absorption) [um^2] of these particles.
    In case the absorption coefficient (mu_abs) [mm-1] of these particles
    is known, adjust sigma_abs [um^2] using the concentration found above
```

```
    until the right mu_abs-value is reached.
    The mu_abs of the layer as-such can be inserted also (see below) !

REPEAT this input for all other types of scatterers!
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-

FOR EACH LAYER AND OBJECT   [ First all layers, then all objects  ]
                            [ ALL entries on separate lines !!!   ]
                            [ All dimensions etc. in mm !!!       ]

  CONVENTION for the numbering of objects:
    objects may be nested (wholly or partially) within each other.
  To determine what properties will be valid in the region of intersection,
    the rule is that there the properties of the object with the LOWEST
    number will be used.
  Example: two overlapping spheres 1 and 2,
    with radii (1) 5.0 mm and (2) 4.0 mm:
    in the region of intersection the properties of (1) will be used.

==>  Following data have to be entered for each layer and object,
     each on a separate line  <==

>> ** LAYER # **    OR      ** OBJECT # **         (#: nr.)

>> (if object) : TYPE of object:
     B: rectangular,        S: spherical,        C: conical,
     R: random spheres in a cylindrical volume (axis // Z-axis),
     M: mirror plane, or:   X,Y,Z: cylindrical, in x,y,z-direction,
     T: half-torus in xz-plane, symm.axis // Y-axis,
     L: lens (thick or thin; symm. axis // Z-axis).
    => IMPORTANT: A change in refractive index of the layer (or object)
     will affect the scatt.cross section (sigma_scatt) [um^2]) of the
     particles in it, because the local wavelength will change.
     Since the program calculates the scattering coefficient mu_scatt
     [mm^-1] from the cross section [um^2] * concentration [mm^-3],
     the scattering coefficient will be affected also.
     You may stick to the "old" value for the scatt.coefficient
     by correcting the concentration.

  FOR EACH TYPE OF SCATTERER IN THIS LAYER OR object:
>>>   CONCENTRATION [in 10^6 per mm^3]
>>>   (if conc.<>0) : VELOCITY [mm/s]
>>>   (if conc.<>0 and velocity<>0) :
    > DIRECTION: X,Y,Z or R (R=random) or Q (= along axis of Q-object)
    > TYPE: Uniform/Poiseuille/Gaussian (enter: U/P/ or 0..9) (see below)

  FOR EACH LAYER OR OBJECT (continuing):
>> REFRACTIVE INDEX of the layer or object (= the medium for the particles;
   if Mirror with 100 % -reflection: enter: 10000).
>> ABSORPTION COEFFICIENT mu_abs [mm-1] of the material of
   The layer or  object.
>> (if layer): THICKNESS [mm]
>> (all objects, except L): no. of HIGHEST layer, in which
   the object extends;
>> (if B-object): dimensions: Xmin, Xmax, Ymin, Ymax, Zmin, Zmax [mm];
```

```
>> (if X,Y,Z,M,Q,R,C-object): nr. of DEEPEST LAYER, in which the object
extends;
>> (if X,Y,Z,M,Q,S,C,T-object): X, Y and Z-coord. of a point on the axis
[mm];
>> (if X,Y,Z,Q,S,C,T-object): inner radius [mm];
>> (if X,Y,Z,S,C,T-object): repetition distance between objects [mm];
>> (if M,Q-object): direction angles: theta and phi [degr.];
         with theta and phi = polar (with Z-axis and azimuthal (with X-axis)
         angles. NB. Pos. Z-axis points INWARD layer system !
>> (if R-object): random spheres in a cylindrical volume:
   >>  X and Y-coord. of a point on the cylinder axis;
       (NB. the cylinder will stretch over the whole of the layer(s) it is
in!)
   >>  concentration [# spheres/mm^3];
   >>  average sphere radius [mm] and cylinder radius [mm];
   >>  standard deviation of sphere radii [mm];
   >>  overlap of spheres allowed? Y/N;
>> (if T-object): (NB. T-object = half torus!)
   >>  Direction of U-shaped opening of half torus:
       towards +Z (enter: +1)  or -Z (enter: =-1).
       (NB. Torus MUST be completely located in 1 (one) layer;
            so: no extension across layer boundaries!!)
>> (if Z,Q,T-object): INNER WALL = specular reflecting (mirror) or absorbing:
       add "100" or "200" resp. to the value of the max.layer nr.
>> (if C-object): Bottom-to-top direction:
       +Z,-Z,+X,-X,+Y,-Y: enter: 1,2,3,4,5,6 resp.
>> (if L-object):
   >>  X- and Y-positions of symmetry axis;
   >>  Z-positions of intersections of front and back surfaces with
symm.axis;
   >>  radii of curvature, front and back (pos/neg if behind/in front of
lens);
   >>  diameter (perpendicular to symm.axis);
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-

> ** CALCULATION MODES & DETECTION **

> = CALCULATION MODES: OPTIONS:   [<...> = averaged over photons] : Enter:
     S:     Reflection:   store <scatt.depths> and max.photon.depth;
     R:     Reflection:   store <scatt.depths> and <Doppl.scatt.depths>;
     T:     Transmission: store: as with R;
     I,J,K: Internal:     store depth of detection  and max.photon.depth,
               = detect at passage (in->out) of a sphere surface
                 or a curved surface of a cylinderical tube or cone;
                   I = all; J/K = at reflection/transmission side;
     A:     Absorption:   store depth of absorption and max.photon.depth;
     E:     All:          = S+T+A   (<version 2018: R+T+A);
     Additionally with all options:
                   store positions immediately after N-th flight-time step
                   (see below, with N>0 and flight time step size dt>0).
     Extra option:
     D:     Duration:     photon terminated immediately after (N+1)th time
step.

  = (Optional) PATH LENGTHS: Geometrical=default. Optical? Add: 0.
  = (Optional) FLIGHT TIME POSITIONS :  (see also in chapter 15.)
```

```
    If storage of flight time positions wanted: add nr. of time steps N
    (max N=9) and time step size dt (in sec.) on the same line with
    S/R/A/T/D/E:
    FORMAT (keep exact!!) (e.g.): if you want:
           Reflection S, max.depth, and optical path lengths,
           with N=5 and dt=1.2E-15 sec: --> enter:   SO+5x1.2E-15

  NB. If CALCULATION MODE = D (Duration) then *.FOx-file (x=0..9)
      contains time-of-flight positions after (N+1)-th time step,
      thus NOT at time of detection or  absorption.

  NB. If CALCULATION MODE = I,J,K (Internal detection at a SPHERE surface
      or curved surface of a CYLINDER or a CONE),
      the program automatically stores the detection positions in
      rectangular coordinates. However, for SPHERES and CYLINDERS
      the program offers optional recalculation of the X,Y and
      Z-positions of photons at their detection position into:
      - polar angle theta (w.r.t. local Z'-axis through sphere center)
      - azimuthal angle phi (w.r.t. local X'-axis)
      This option is available when plotting or with Ascii-output of
      the photons.
        I = all;
        J = photons, internally detected at reflection side only
        K = id. ... at transmission side.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
-

DETECTION FIELD AND NUMERICAL APERTURE OF DETECTORS

DETECTION FIELD [mm]:
> Min.Xfield, max.Xfield, min.Yfield, max.Yfield [on 4 separate lines].
    (NB: for 2- and 3-dim storage, this field will be divided
         in 20x20 pixels in the X- and Y-direction.
         In the "scattering" mode a detector with arbitrary dimensions
         may be placed in this field; division in 20 pixels.)
    In case you want ring-shaped detectors:
         consider Xfield as Rfield, and insert min.Yfield=max.Yfield=0;
         then min.Xfield<max.Xfield is necessary! otherwise: no width!
    In case of absorption followed by fluorescence/Raman:
         use identical detection fields in both cases!!

> (on one line!:) NUMERICAL APERTURE (NA) of detector (IN VACUO !!) +
        REFRACTIVE INDEX OF DETECTOR SURFACE;   (format example: see below.)
    NA = sin(th), with th = half opening angle (0<th<=90 degr.);
    (completely open: NA = 1)
    - In case of reflection and detection at interface of layers
      N and N+1: (NB. surface: N=0) :    -> add N*10 to NA.
    - In case of internal detection upon passage of the curved surface of a
      sphere, cone, torus or cylinder :  -> add object.nr*10 to NA
    EXAMPLES OF INSERT FORMAT :
    with NA=0.73 and refr.index of detector = 1.58:
    - for detection at surface (layer 0) -> enter: 0.73/1.58
    - for detection at interface of layers 2 and 3 -> enter: 2.73/1.58
    - for detection in object 3  -> enter: 3.73/1.58
    - if Calculation mode = T (transmission) -> enter 0.73/1.58.

> ** DEPTH PIXELS **
```

Each layer may be divided into depth pixels. Min. 1 pixel/layer.
Useful for 2D/3D-output files.
> For each layer: enter nr. of pixels [on separate lines]



\- -

Fig.6. Photon path tracking: Photon “bananas” arising by scattering from beam entrance point to exit area (between 5 and 6 mm). For clarity, all photon paths were rotated afterwards as if the photons had emerged on the +X-axis.

> ** FILE OUTPUT **
(NB. The output files will have the same name as the *.INP input file)
> DIRECTORY (FOLDER) to store the output files.
> STORE : *.FOT-file(s) (enter F), *.2D/3D-files (enter D) or both (enter B)
(only if PATH TRACKING wanted; otherwise skip): Add to F/D or B:
A: Store positions of all scattering events (+reflections/refractions);
this option means spacing of about 1/mu_s;
R: Store # scatt.events * (1-g) (+refl./refr.);
this option means spacing of about 1/mu_s' with mu_s'=mu_s*(1-g);
example: this line should read e.g.: FA or F or BR or B or ....
> (only if Store = D or B :) Linear/Quadratic XY-PIXELS in 2D/3D-files:
enter L/Q

> ** NR. OF PHOTONS ** NB. The number of photons is only restricted
through the available space in the output directory.
However, when analyzing the data with this program, the number of
photons may be restricted due to the available RAM-space.
The 2- and 3-dim-files contain information of ALL photons.
To CHANGE nr. of photons DURING simulation:

```
        - Interrupt simulation using "Interrupt"-button;
        - change Nr. of photons in *.INP-file;
        - Resume simulation
> NUMBER OF PHOTONS, to be : either:
  - emitted by the source  (enter E),
  - injected in the sample (enter I),
  - accumulated in the detection window (enter D),
  Enter: E,I, or D, followed by the number, on same line (e.g.:  D100000).
  NB. With fluorescence/Raman:
      if D chosen, and the nr. of absorbed FOT-photons is not sufficient,
      the absorption FOT-file is used repeatedly (at start
      the photons will have different directions compared with previous use).
> (only if moving scatterers present; otherwise skip):
      Store DOPPLER-SHIFTED photons only? Enter Y/N
```

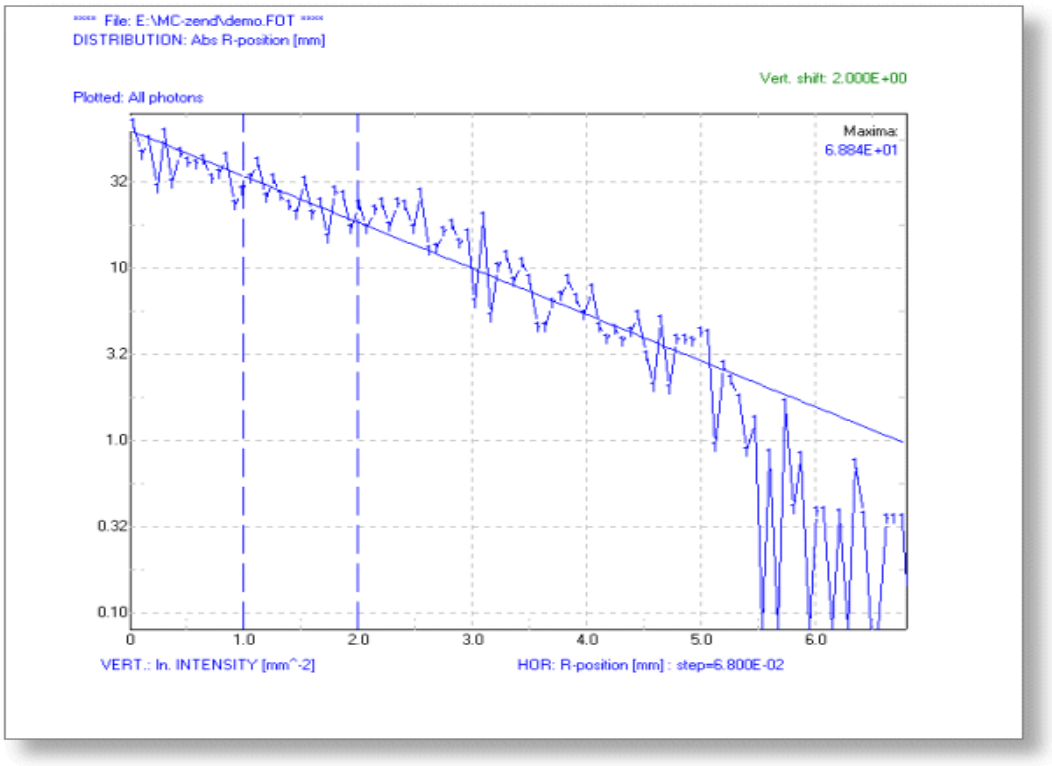


Fig.7. Example of output plots. Here ln (Intensity) vs. R-position from the Z-axis. Also included: a model approximation (solid line)

==== 14. VELOCITIES : profiles and distributions ====

```
Options: (in all cases: velocity to be entered = <v> !!)
- Uniform: all velocities of the present scatterer in the present
           layer or object have the same magnitude;
- Poiseuille: a parabolic profile, extending to the layer or object
           boundaries (where v=0):
     = layer with flow in X or Y-direction; (depth variable: d; thickness D):
             v(d) = vmax.[ 1 - d^2/(D/2)^2];
             vmax = 3/2 * <v> ;  (vmax = v on symm.axis)
             if flow in Z-direction: treated similar, with z = variable;
     = tube: cylinder: (variable: r; radius R):
             v(r) = vmax.[ 1 - (r/R)^2 ]; vmax = 2 * <v> ;
```

```
            torus: similar, flow always parallel to (curved) axis;
     = block (rectangular): approximated as N-dimensional parabolic profile,
               e.g. if flow in X-direction: 2 distributions (N=2)
                    with variables y and z:
            v(y,z) = vmax.[ 1 - y^2/(Y/2)^2].[ 1 - z^2/(Z/2)^2],
                  with Y and Z the dimensions.
          vmax = (3/2)^N * <v>, with N=2; if random direction: N=3.
- Gaussian:  Gaussian profile, e.g. for Brownian motion;
          sigma is expressed in N * 5 % of <v>, with N=0..9;
          N to be entered in the input file.
NB. Objects of type R,S,L (spheres or lenses): only random directions.
```

==== 15. PHOTON DATA STORAGE in *.FOT AND *.PAT-files : ====

*.FOT (FOx)-files:

```
For each photon the following data are stored (24 bytes/photon):
  {Ri/Ii: Real/Integer; i=nr. of bytes}:     (*)= see below: COORD. UNITS
- I2: X-coord. of point of detection or absorption [unit: 1 or 10 um]; (*)
- I2: Y-   ..  ..  ...  ..     ..    ..     ..     [id.];  (*)
- I2: either: average depth of all scatter events; or :
       maximum depth, or: point of detection or absorption [id.];
- I2: id. of all Doppler scatter events [id.];
- R4: path length [mm];
- R4: Doppler frequency [Hz];                        ==> (*): see below <==
- I4: number of scatter events;
- I1: percentage of Doppler scatter events;
- I1: theta: angle with pos.Z-axis (!) at detection or absorption [deg] (*)
- I2: phi: angle with projection of vector on XY-plane with X-axis [deg] (*)
Convention:          :       theta         :        phi             :
- reflection         :  180 >= .. >=  90   :    0 <= .. < 360       :
- transmission       :   90 >= .. >=   0   :    0 <= .. < 360       :
- absorption         :  180 >= .. >=   0   :    0 <= .. < 360 (o)   :
(o) stored in *.FOT-files as: -phi-1  .             ==> (*): see below <==

The photons are stored in the *.FOT-file in bunches of 1000.
```

(*) COORDINATE UNITS

```
For program versions < 2018:
The X-,Y-coordinates are stored as "integers", and the depth-coordinates
as words" in units of 1 (default) or 10 micrometer;
The 10 micron-unit is used:
- for XY-coordinates: in case the XY-detection field extends beyond
  +/- 32.767 mm (= 32767 micron = max. |smallint| value);
- for Z-coordinates: in case the layer system thickness > 65.535 mm
  (= 2 x 32767 micron = max. word value)
  (smallint: 2 bytes:  -32768..+32767;   word: 2 bytes: 0..65535)

For program versions >= 2018:
The X-,Y-coordinates and the depth-coordinates are stored as "singles",
4 bytes each, with no size limitations;
```

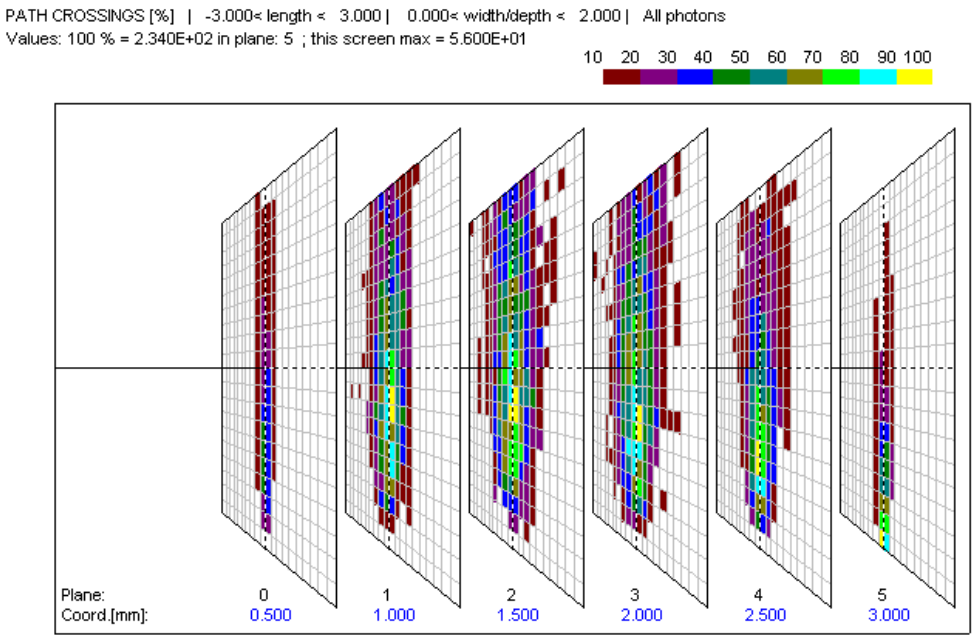


Fig.8. 3D-plot of Path tracking: photon "bananas": average depths of photon paths. Entrance at position 0; photons emerging between positions 5 and 6 mm from entrance.

In case of INTERNAL DETECTION (calculation mode= I,J,K):
The POSITION coordinates of the point-of-detection are stored as follows:
: X-, Y- and Z-coordinates on layer system axes;
: and the depth-coordinate as above, all in integer format;
In case of recalculation from Carthesian into angular coordinates (see above) here the polar axis is the local Z'-axis (// Z-axis) through the sphere center or the symmetry axis of the cylinder or cone , and the azimuthal angle is measured from the local X'-axis (// X-axis).
The DIRECTIONAL theta and phi, otherwise measured as mentioned above, now are measured from the local coordinate system in the detection point (local Z"-axis = polar axis = normal through sphere center, and local X"- and Y"-axis = south-north and east-west direction on sphere).

FLIGHT TIME POSITIONS:

Detected photons can be followed during flight; stored in *.FOx files, with x=0..9 time steps, with time-step size to be inserted in the *.INP-file.
*.FOx files have same format as *.FOT-files.

Using the option: "View structure of layer system" the evolution of the photon density can be followed on-screen. n.

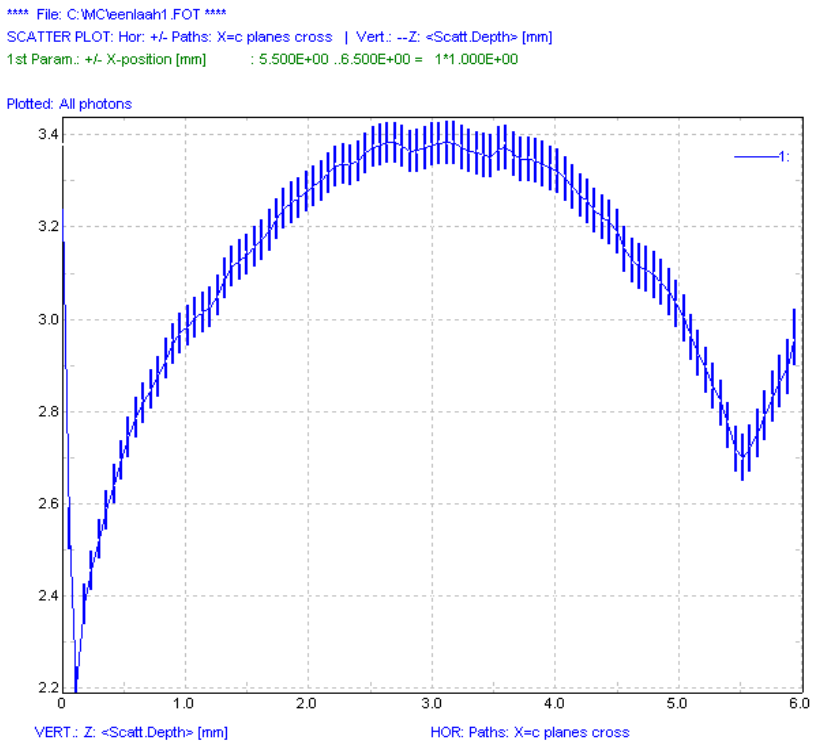


Fig. 9. Paths tracking: Averaged depths of photons, with standard deviation in the average. Settings: see fig. 5. Plot for distance > 5.5 mm is due to spurious photons.

PATH TRACKING : PHOTON PATHS STORAGE IN *.PAT-FILES (path tracking)

Only paths of detected photons are written in the file.
The *.PAT-files have similar format for x,y and z-data as the *.FOT-files.
This means that each path position is written as:
"smallint, smallint, word" for x,y, and z respectively (see above).
Each photon path is written as follows:
- leader: x=9999, y=9999, z=N, with N = nr. of events to follow
(scattering/reflection/refraction)
- for each of N events: x, y and z-value (in units as mentioned above).

==== 16. AVERAGED DATA : in pixels in detection field ====

In order to have data about all photons available, averaged data may be stored in *.2D and *.3D files, in which 2- and 3-dimensional pixels are filled during the calculations.
The horizontal (X and Y) pixels are defined as amounting to 1/20 of the dimensions of the detection field; the vertical (Z) pixels have to be entered in the *.INP file separately (see below).

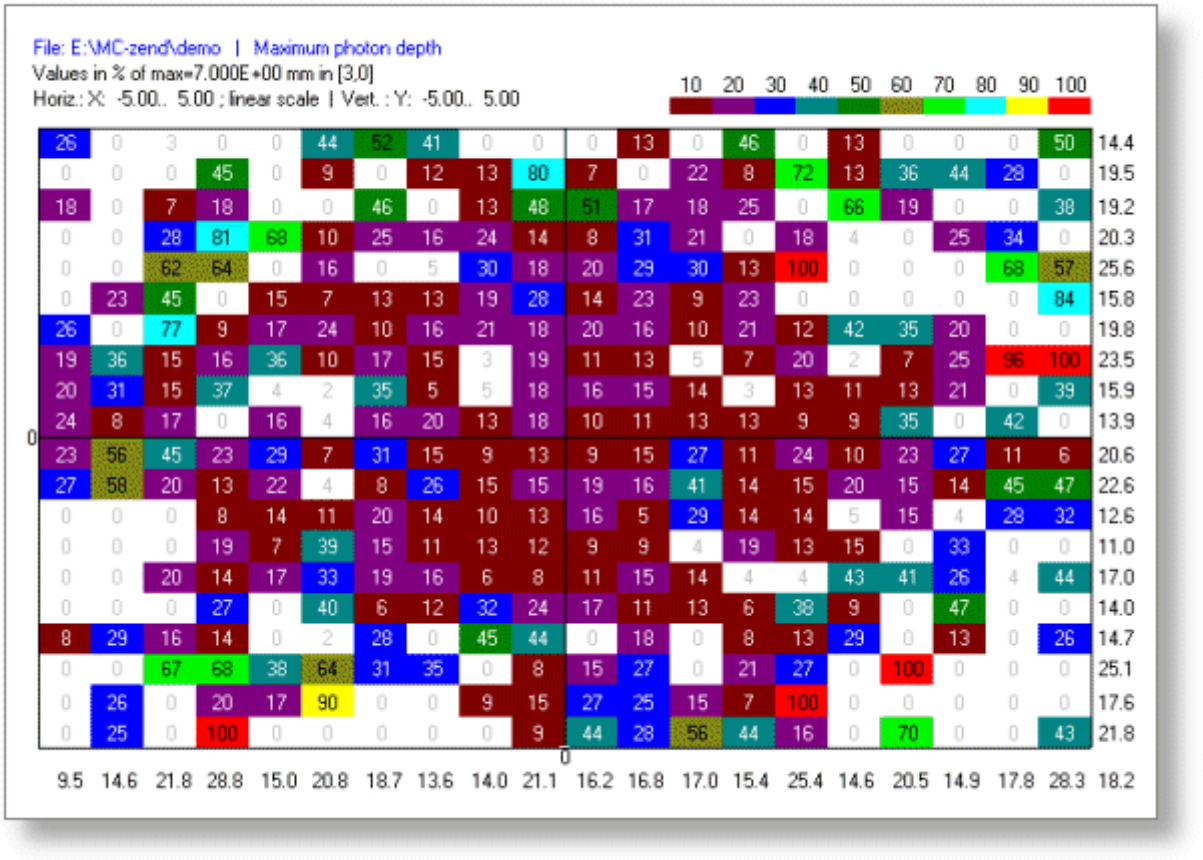


Fig.10. Example of 2D-plot: Here the maximum photon depth plotted as function of (x,y)-positio

==== 17. INTENSITY APPROXIMATIONS : I(r) (r= source-detector distance) ====

```
The Option "Results" + "Distributions" + "Hor.Var.= r"
offers the possibility to fit some
analytic intensity models to the simulated intensity distribution.
The models are:
- I(r) = c1 * exp(-c2*r) / r^x , with x = 0, 0.5, 1 or 2.
- I(r) = according to the Farrell-model (with 1 source).
Here the constants are:
- c2 = mu_eff = sqrt( 3*mu_a*(mu_a+mu_s')), with mu_s'=mu_s*(1-g);
       (mu_a, mu_s = absorption and scattering coefficient in 1/mm;
        mu_s' = reduced scatt.coeff.; g = anisotropy factor);
- c1 = power*mu_eff^2/(4*pi*mu_a), with power = nr. injected photons;
       (NB. this expression for c1 is valid for x=1 only !)
The Farrell model is:
I(r) = albedo*power/(4*pi) * [z01*(mu_eff+1/r1)*exp(-mu_eff*r1)/r1^2) +
                        z02*(mu_eff+1/r2)*exp(-mu_eff*r2)/r2^2) ],
with z01 = 3D; D= 1/(3*(mu_a+mu_s');  z02 = z01+4AD;
     r1^2 = z01^2+r^2; r2^2 = z02^2+r^2;   albedo = mu_s'/(mu_a+mu_s')
     A  = Fresnel correction: A = (1+ri)/(1-ri);
     ri = 0.668 + 0.0636*n + 0.7099/n - 1.4399/n^2;
     n  = relative refract.index from scattering medium to detector medium.
Output: c1 and c2, or (for Farrell) mu_a and mu_eff, resulting chi-squared.
```

NB. The circular integral (CI) over the Farrell-distribution:
CI = int{ I(r)*2*pi*r*dr , r=0..infinity } gives:
= albedo/2 *power*exp(-del)*[1+exp(-4A*del/3)], with del= mu_eff*z01;
If "power"=# injected photons, then CI should be = # reflected photons.

==== 18. PHOTOACOUSTICS : DETAILS ====

The program calculates the PA-RESPONSE on the basis of a spherical source with dimensions of an absorption voxel. This implies a bipolar shape, given by : (c^3)*k * exp [- (ck)^2], with k =(i-i0)*dt, (i - i0)*dt is the time difference to the time point of the zero-crossing (at i0*dt) of the bipolar shape, i = nr. time steps. The constant c is: c = 1/[sqrt(2)*sigma] = 1/ tau_e, and sigma (or tau_e) is the characteristic width (see: TIME REGIME). This bipolar shape is then multiplied by:

(1) exp(-mu_att * r) / r ,
where r = the distance from absorption voxel to detection pixel, and mu_att is the ultrasound attenuation coefficient (averaged over the frequency band of the acoustic signal),
(2) the area of a detector element [mm^2].

TIME REGIME:

Here the time window (nr. of time elements i and element size dt) can be inserted. The time-of-flights to be calculated are: perpendicular and diagonally through the (reduced) absorption volume. A measure for the PA pulse width is tau_e: the "effective pulse length", given by the laser pulse HWHM and the transit time of the acoustic pulse through the elementary source ("radius" =Z-vox./2).

tau_e = sqrt [HWHM^2 + (Z-vox./2/sound.velocity)^2].

which defines sigma: tau_e=sigma*sqrt(2).
The peak-peak time (from pos. to neg. peak) = 2*sigma.
We take i0 to begin at 3* sigma before the zero-crossing of the signal.
The starting time of the sound recording can be shifted, in order to avoid time regions where no signal is expected.

GROUPING OF DETECTORS:

The PA-response is calculated for each detector at its center point. This means that no interference of sound waves over the surface is taken into account. To do that properly, it is advised to use small detector elements and GROUP those into X- and Y-groups. In doing so, probable interference is calculated automatically.
Each group MUST consist of an integer nr. of detector elements.

DIRECTIVITY OF DETECTORS:

Each detector "looks approximately forward" from its center point into the absorption field of voxels,
but can be given a NUMERICAL APERTURE (called "directivity").
Directivity correction: relative w.r.t. 0 degrees;
- available types: U: Uniform, G; Gaussian, T: triangular;
- HWHM (half-width at half-maximum to be entered in degrees.
(normalisation: sensitivity = 1 for perpendicular view (0 degr))

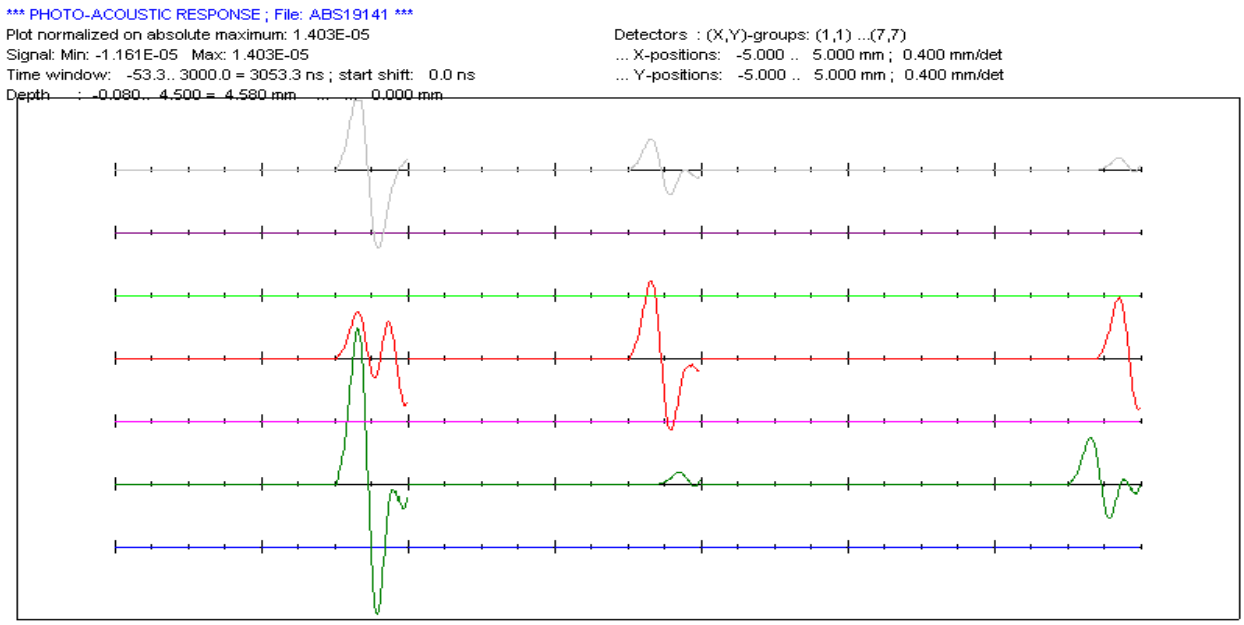


Fig. 11. Photoacoustic response at a 7 x 7 - detector array of a sample consisting of several absorbing objects in a scattering (but not absorbing) medium.

==== 19. FREQUENCY-MODULATED PHOTON SOURCE ====

Normally the photons are injected as a delta-pulse in time, all at t=0.
The path length distribution is similar to the time-of-flight distribution;
the prop.factor is 1/light velocity.
The program has two options for recording the paths:
(a) geometrical or (b) optical.
In (a) the prop.factor is taken as 1/light velocity in the 1st layer;
in (b) the optical path = geometrical path / local refractive index;
therefore here the prop.factor = 1/light velocity on vacuo.

However, the photons could also be injected using a frequency-modulated light source, with typical frequencies of 100 MHz - 2 GHz. Then the frequency response is the Fourier transform of the time-of-flight distribution upon the pulse input.
The plot output offers options for plot of the phase delay (in degr.) and the modulation depth (dim.less), defined as:

phase delay(n) = arctan { Im[f(n)]/Re[f(n)] } ,

modulation depth = sqrt{Im^2[f(n)]+Re^2[f(n)]} / sqrt{Re^2[f(0)]} ,

where f(n) is the (Fast-) Fourier transform of the time-of-flight distribution.
If time step = dt, then max.freq. (f_max) = 1/(2*dt).
The freq.step df is f_max/(N/2), where
N = nr of time steps in the time-of-flight distribution,
replenished up to a power of 2 (e.g. N = 512, 1024, 2048).

The program also offers the possibility to calculate the phase delay and modulation depth using analytic models published by:
- Haskell et al., for 1-layer samples,
- Kienle et al., for 2-layer samples.

References:
- R.C. Haskell et al., J.Opt.Soc.Am. A 11, oct 1994, p 2727-2741.
- A. Kienle et al., Appl.Optics 37, Feb 1998, 779-791.

==== 20. RUN TIME ERROR MESSAGES (from Borland Pascal) ====

1 Invalid function number
2 File not found
3 Path not found
4 Too many open files
5 File access denied
6 Invalid file handle
12 Invalid file access code
15 Invalid drive number
16 Cannot remove current directory
17 Cannot rename across drives
100 Disk read error
101 Disk write error
102 File not assigned
103 File not open
104 File not open for input
105 File not open for output
106 Invalid numeric format
200 Division by zero
201 Range check error
202 Stack overflow error
203 Heap overflow error
204 Invalid pointer operation
205 Floating point overflow
206 Floating point underflow
207 Invalid floating point operation
210 Object not initialised
211 Call to abstract method
212 Stream registration error
213 Collection index out of range
214 Collection overflow error
215 Arithmetic overflow error
216 General protection fault

Stub errors:
Stub error (0001) needs at least 286
Stub error (2002) can't find rtm.exe
Stub error (2003) can't find DPMI16BI.OVL
Stub error (0012) file not found
Stub error (0013) path not found
Stub error (0015) file access denied
Stub error (0018) not enough memory to load file
Stub error (001A) invalid environment
Stub error (001B) invalid file
Error: no DOS extensions in the DPMI server
Error: needs DOS 3.x or higher
Error: failed to locate DPMI server (DPMI16BI.OVL)

==== 19. Contact address ====

The name of the help file is: MONTCARL.TXT (Ascii-format)

When in trouble, contact:

Dr. ir. F.F.M. de Mul,
e-mail: ffm “at” demul.net

Previously: University of Twente, Department of Applied Physics,

Research Group: Biophysical Technology / Biomedical Optics,
P.O.Box 217, NL - 7500 AE Enschede, the Netherlands,

For regular updates of the program, see www.demul.net/frits

- -

MORE FIGURES:

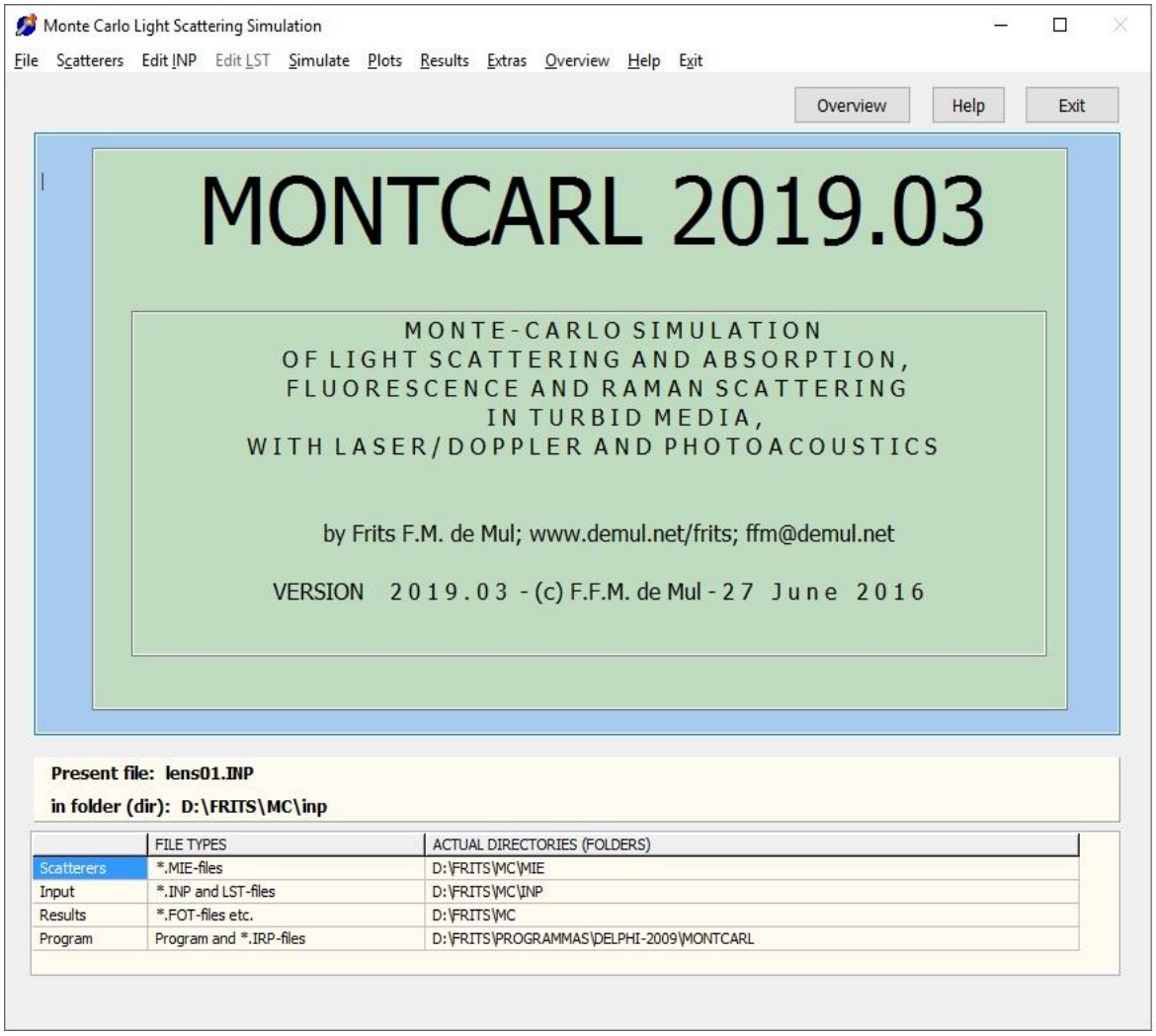


Fig. 12. Menu screen (with Tabs)

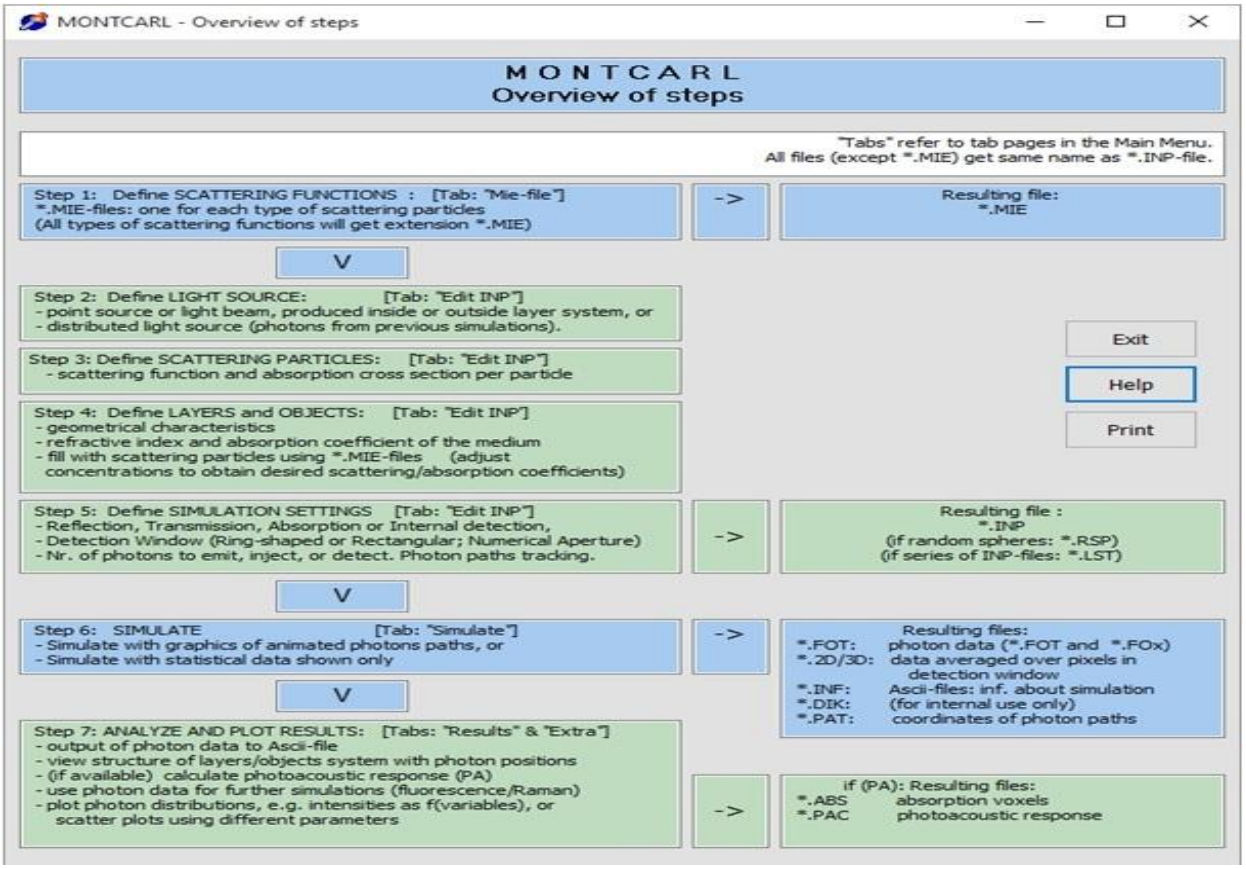


Fig. 13. Overview about how to input settings and to run simulations

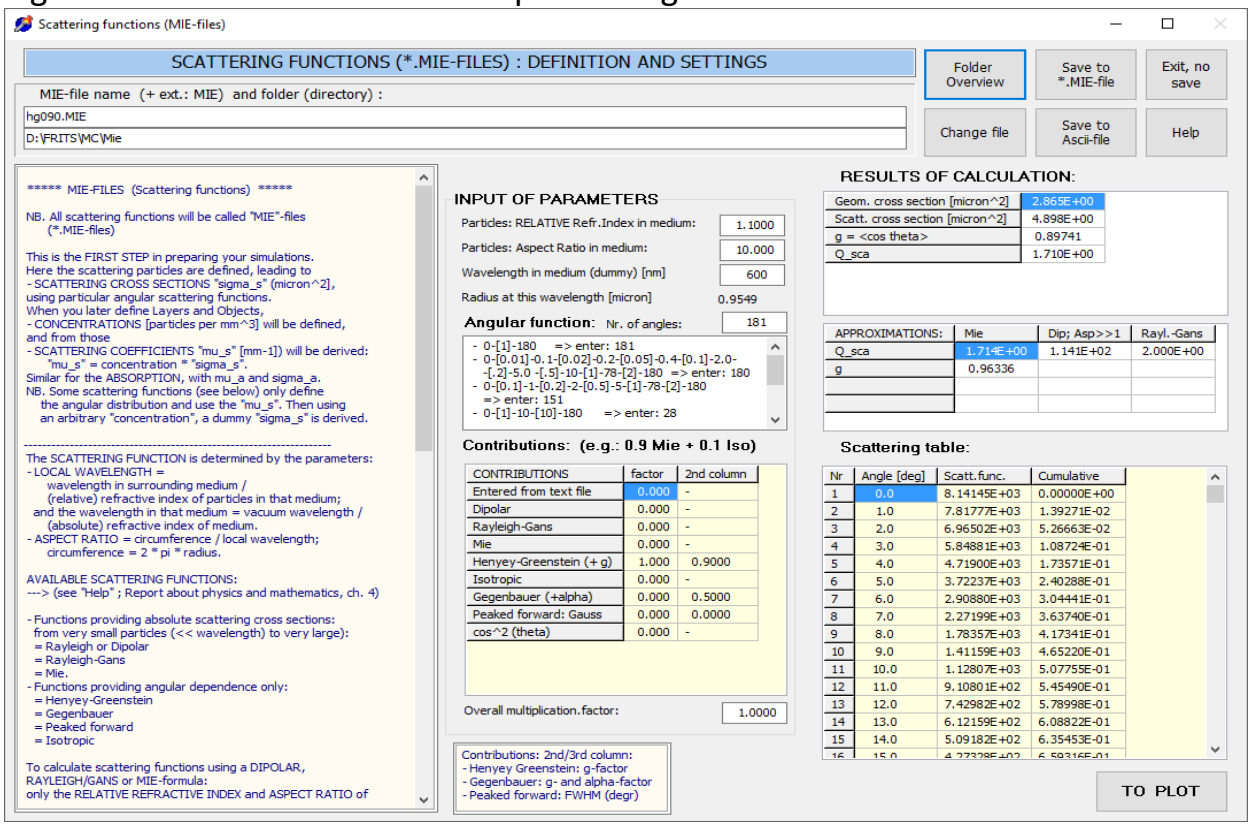


Fig. 14. Creation of scattering functions (called: *.MIE-files)

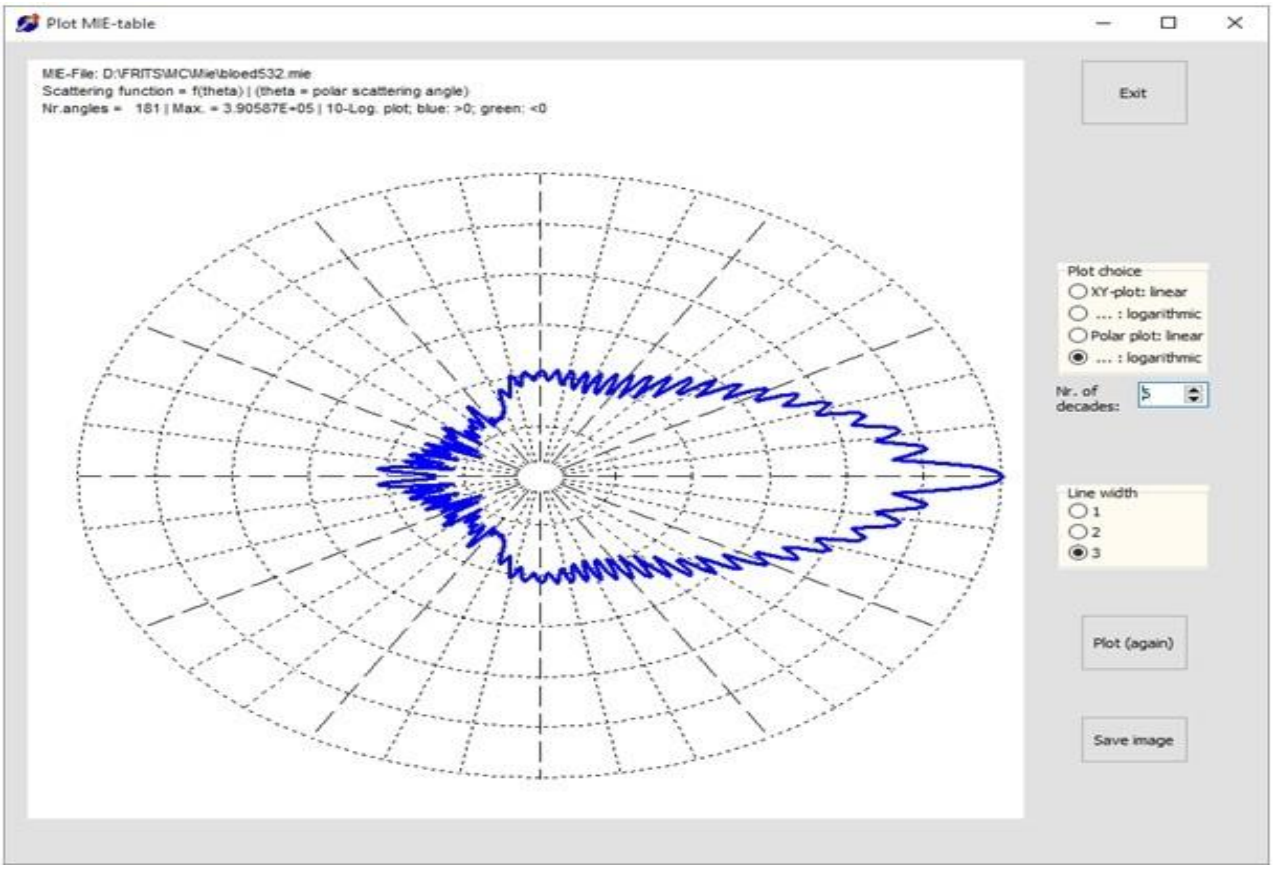


Fig.15. A scattering pattern (combination: Mie + Henyey-Greenstein-functions)

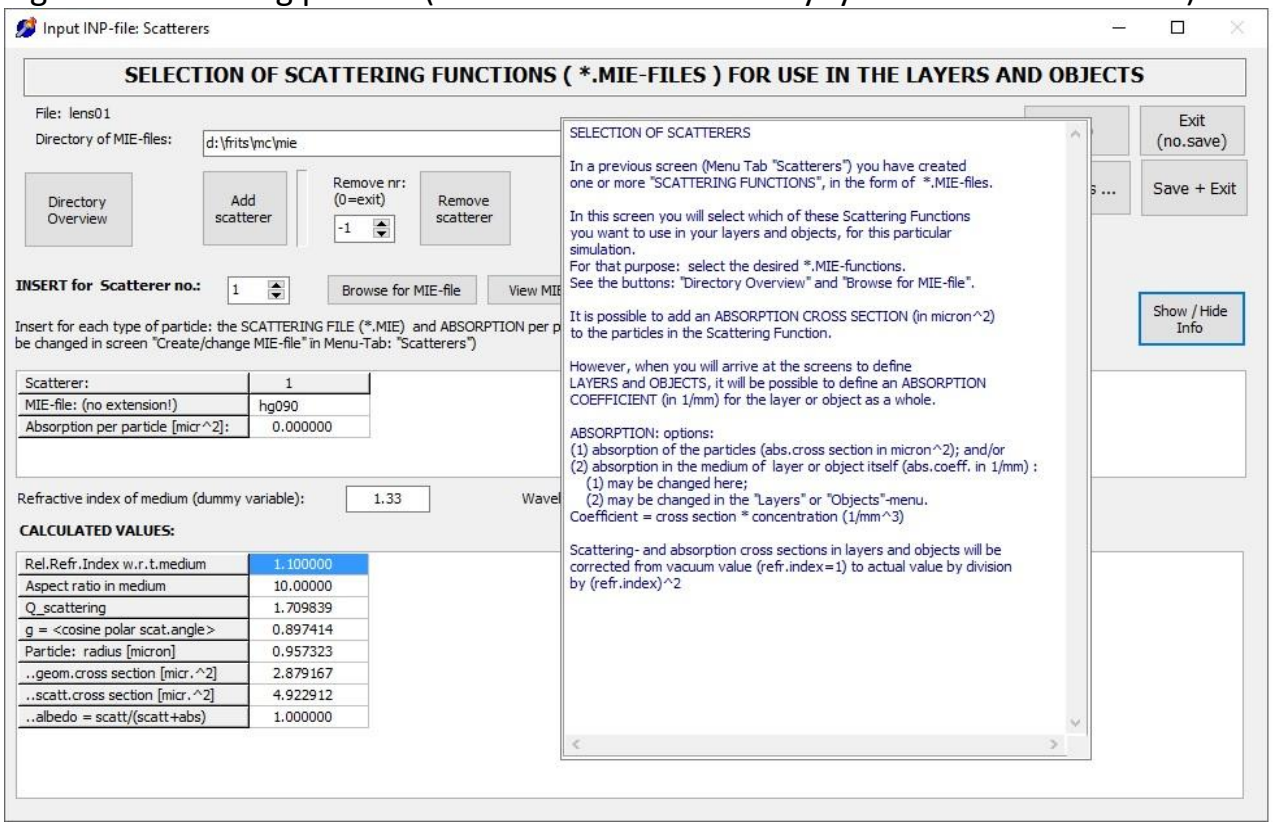


Fig.16. Selection of scattering functions for use in layers and objects

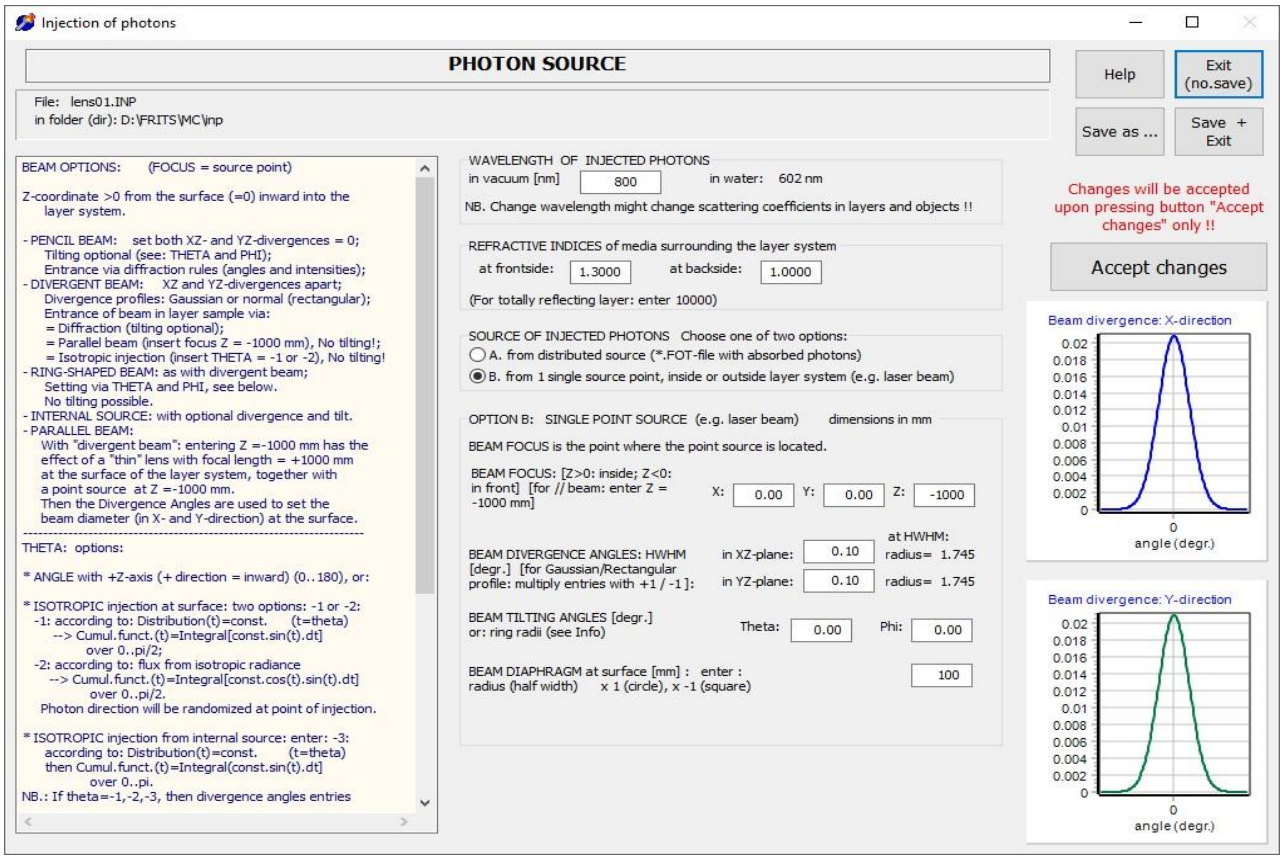


Fig.17. Input of data for the light source (e.g. laser data)

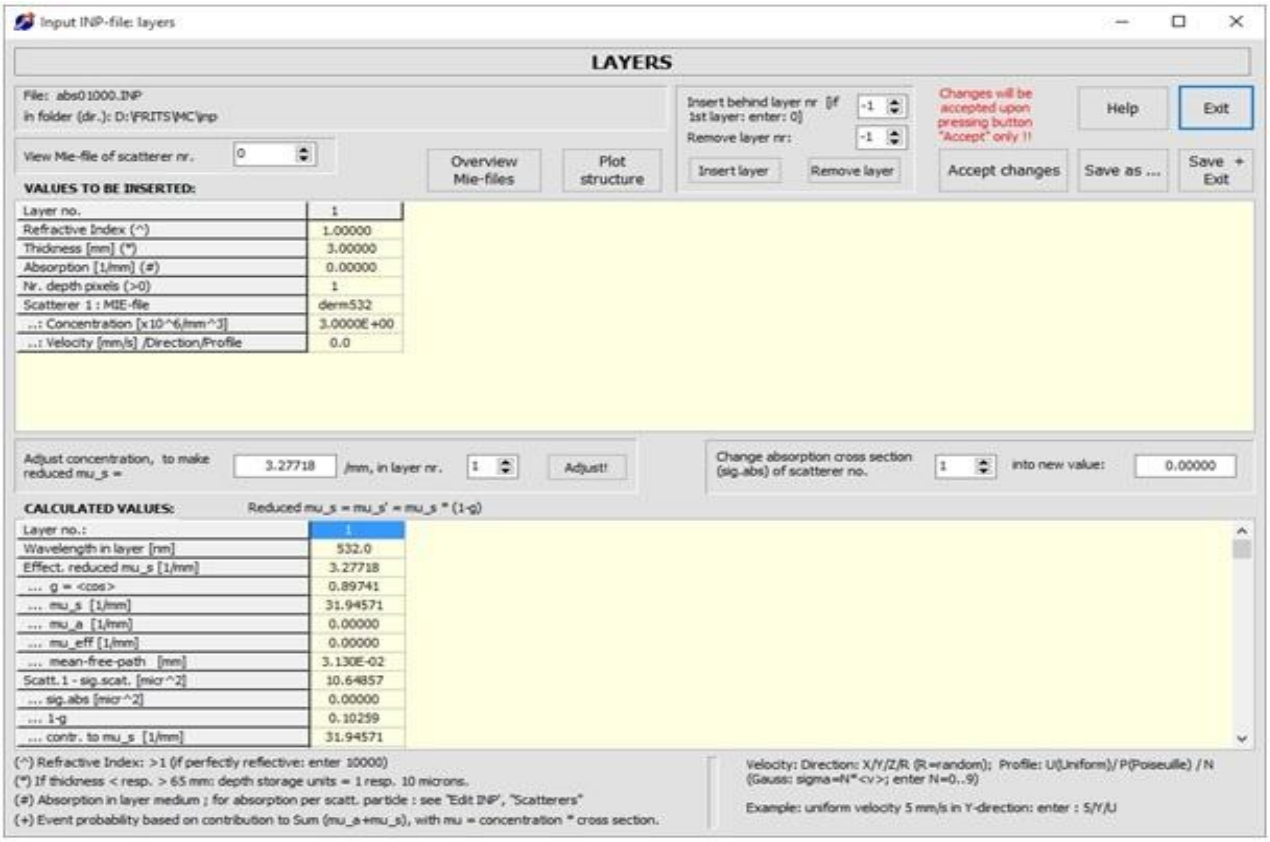


Fig. 18. Input of data for layers (more than 1 layer possible)

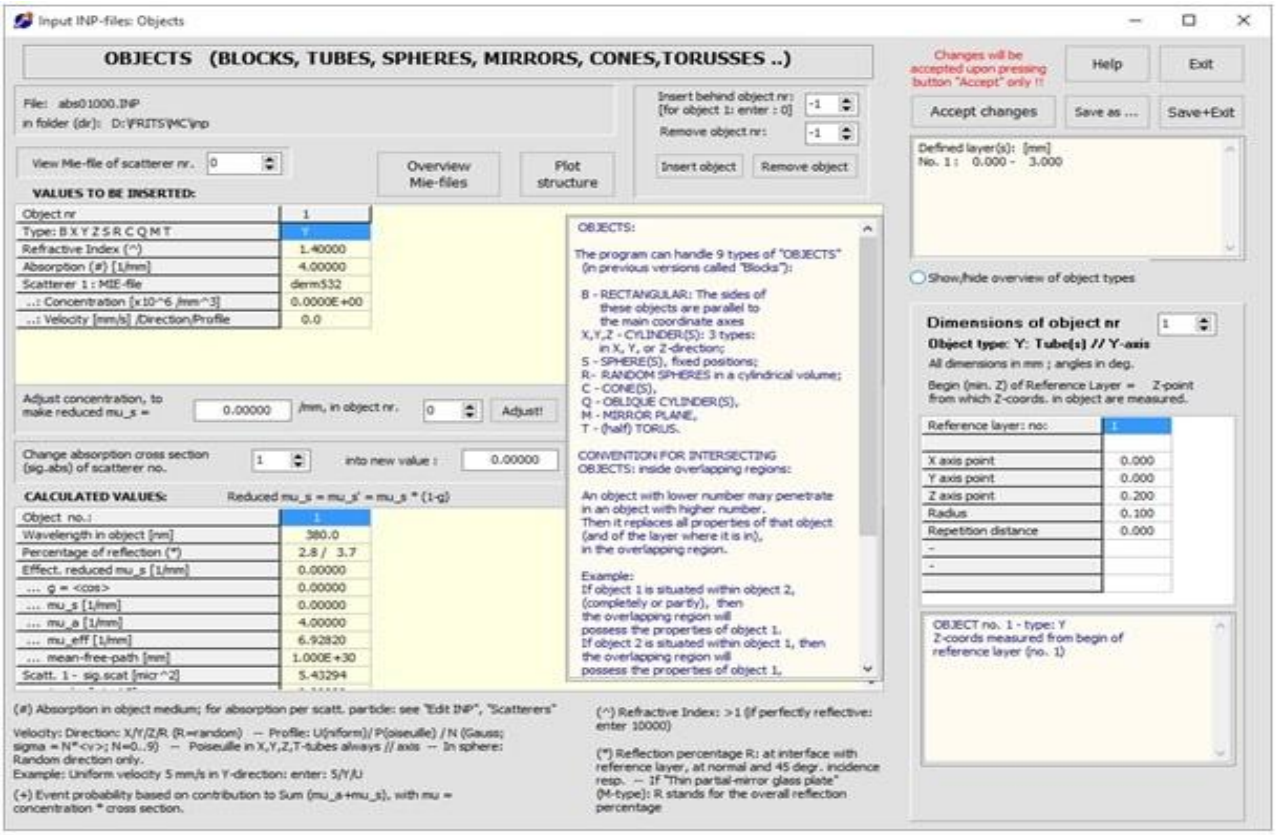


Fig. 19. Input of data for objects (spheres, tubes, mirrors, cones, blocks…)

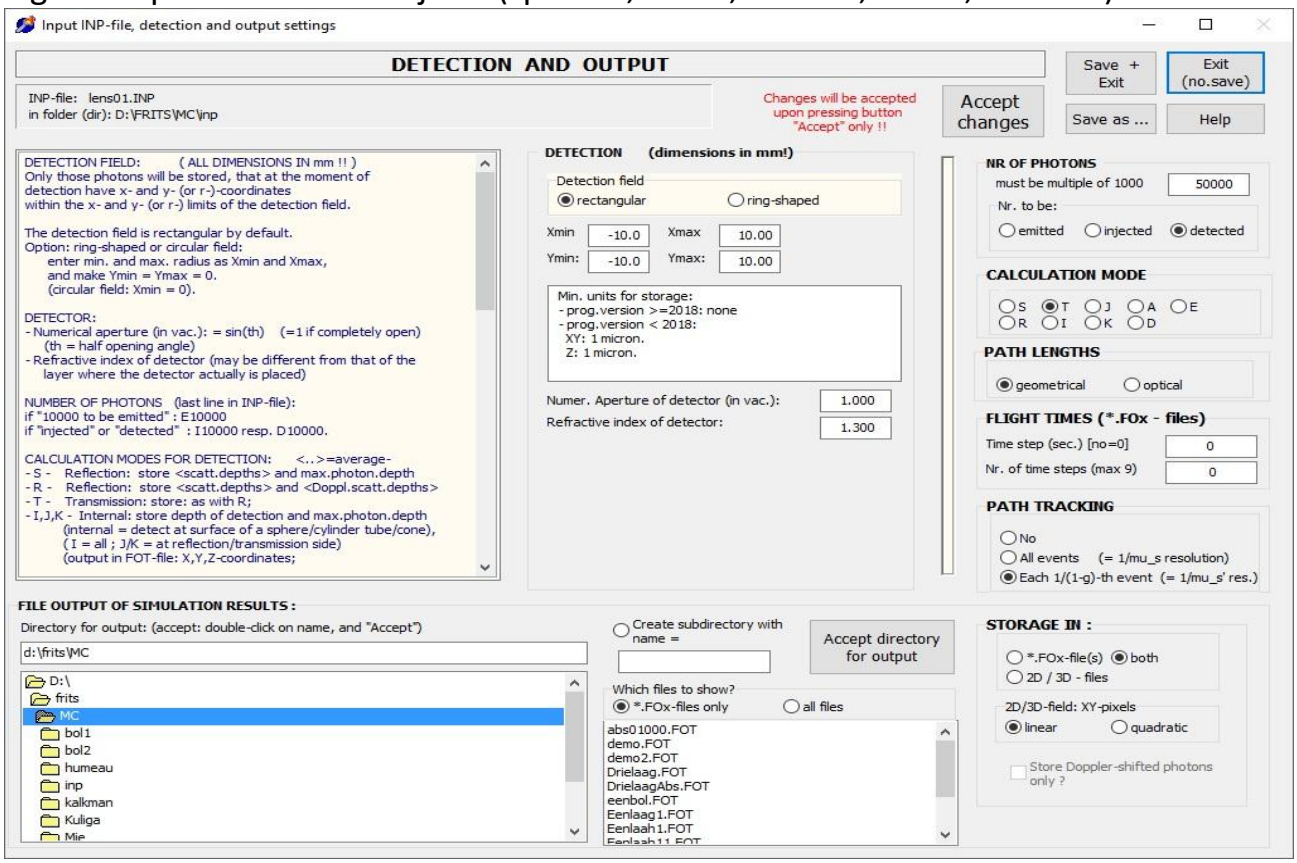


Fig. 20. Detection, calculation mode, flight and path tracking and output

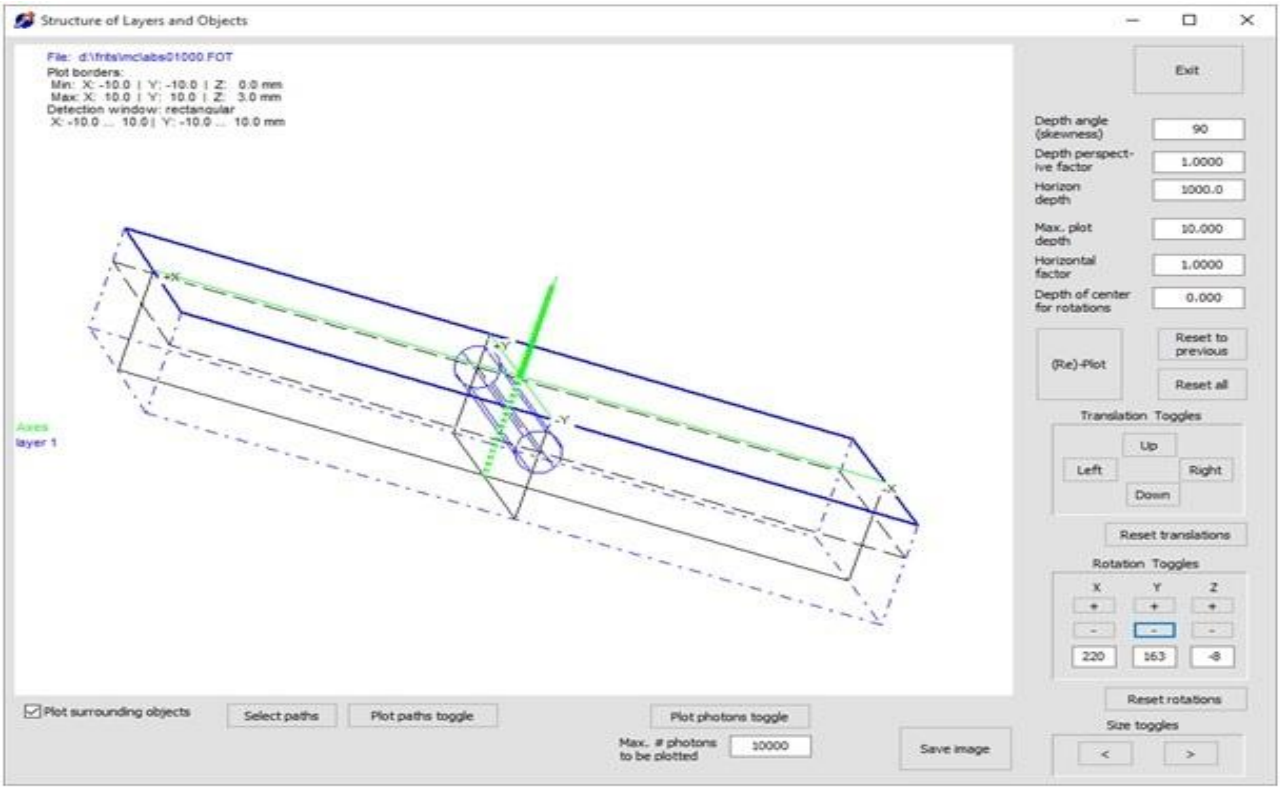


Fig. 11. Structure of the layer system with 1 layer and 1 tube in Y-direction

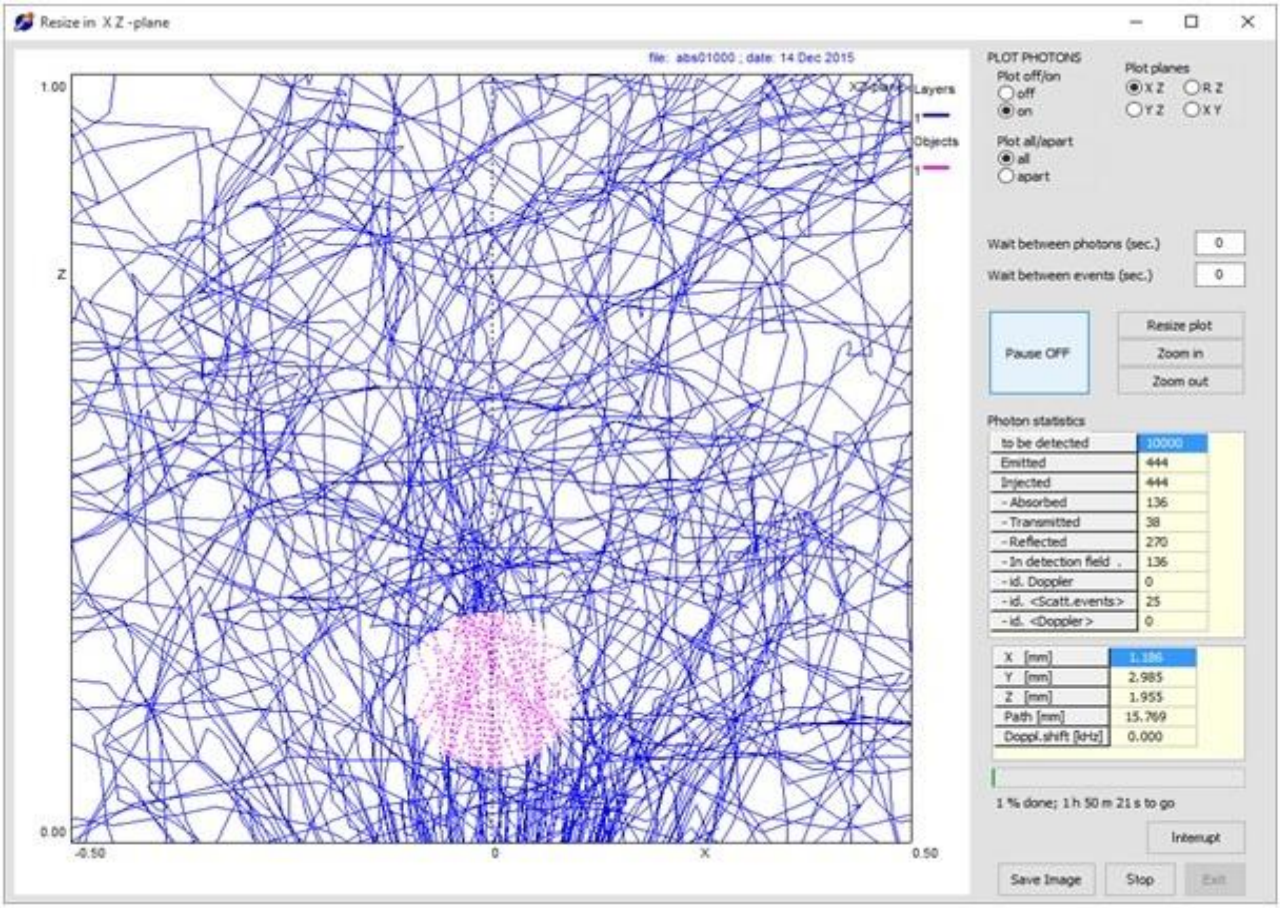


Fig. 22. Simulation of the structure of Fig. 21. View // Y-axis (XZ-plane)

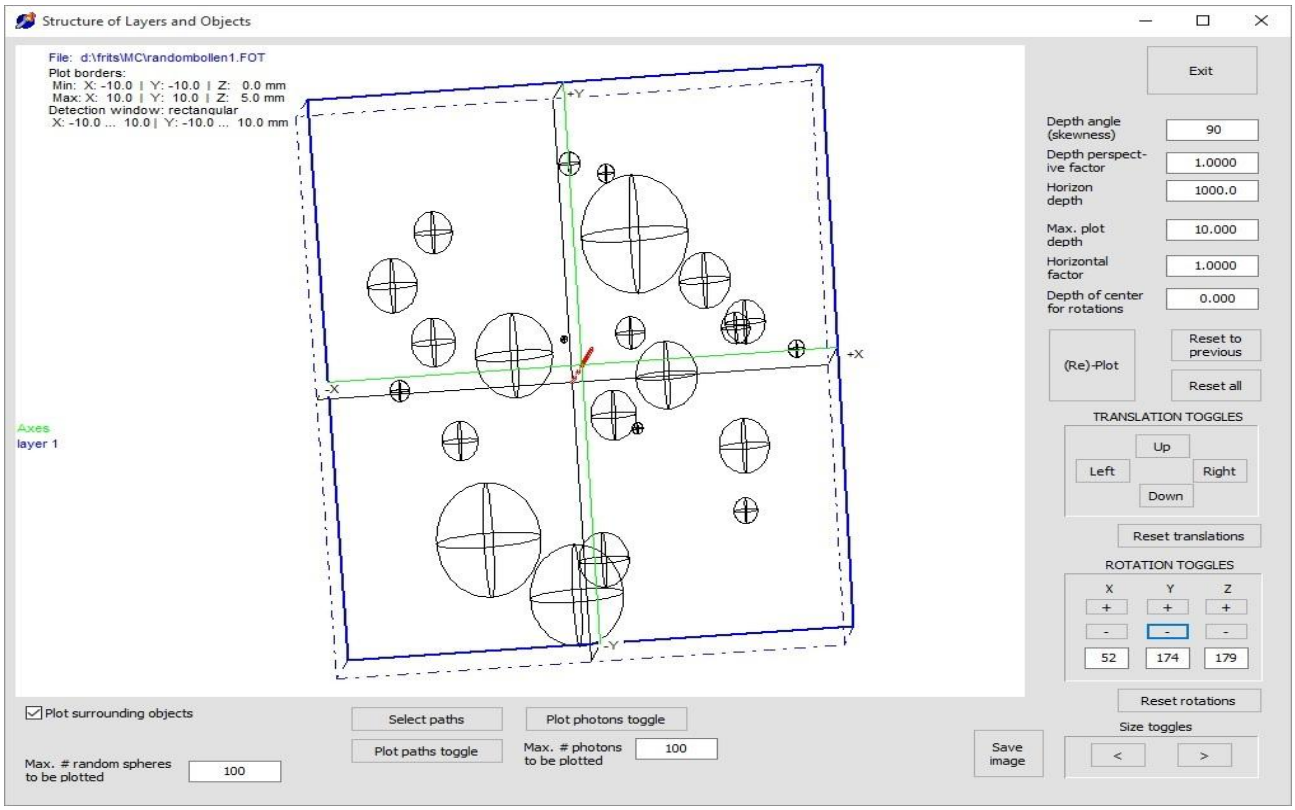


Fig. 23. Structure of a single layer with random spheres

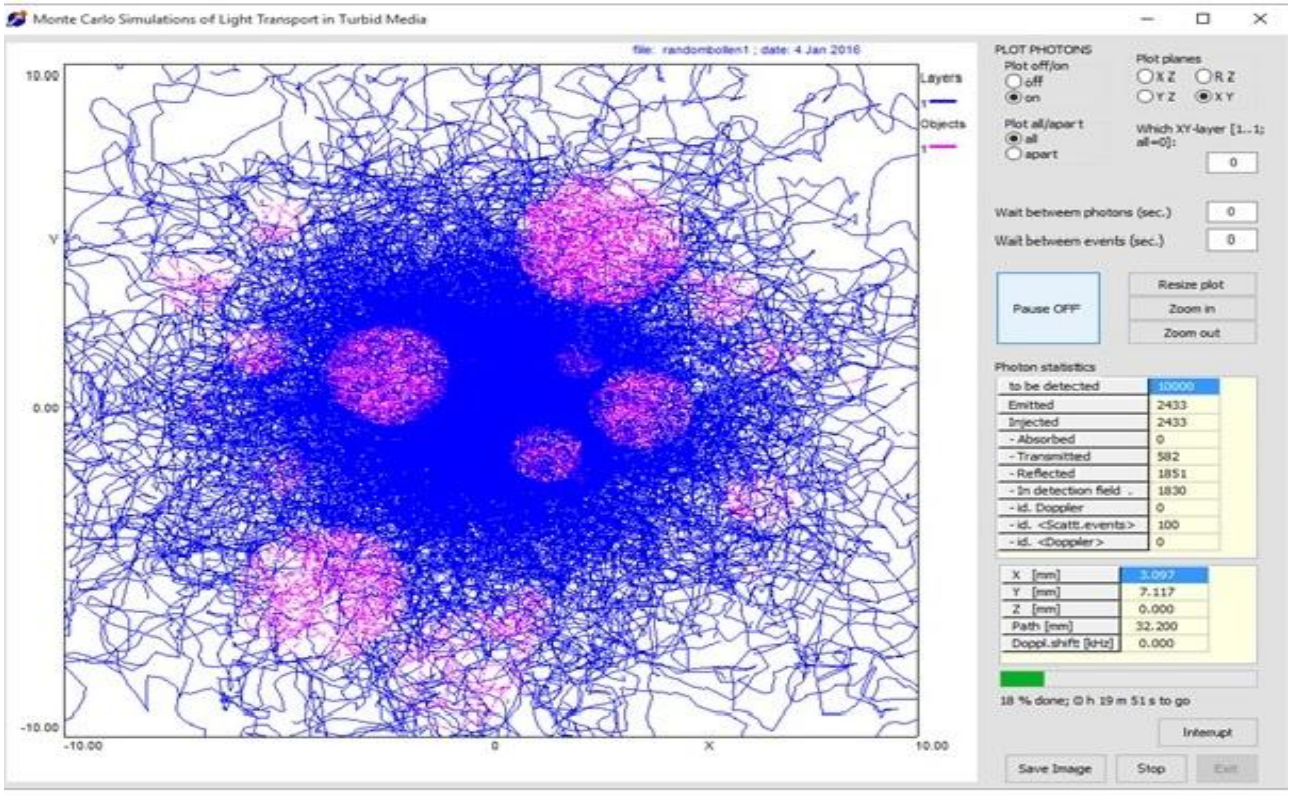


Fig.24. Simulation of the structure of Fig. 23. View // Z-axis (XY-plane)

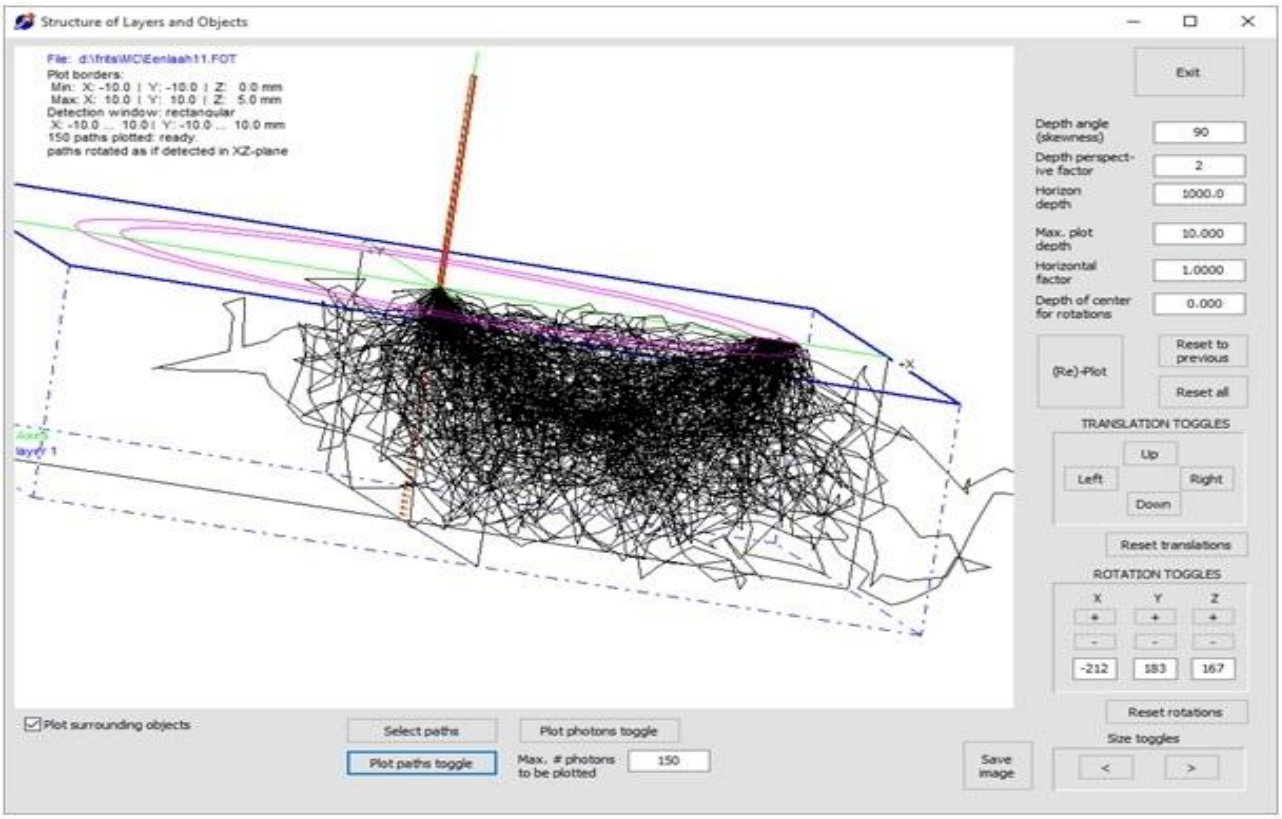


Fig. 25. Path tracking for selected points of emergence. Single layer

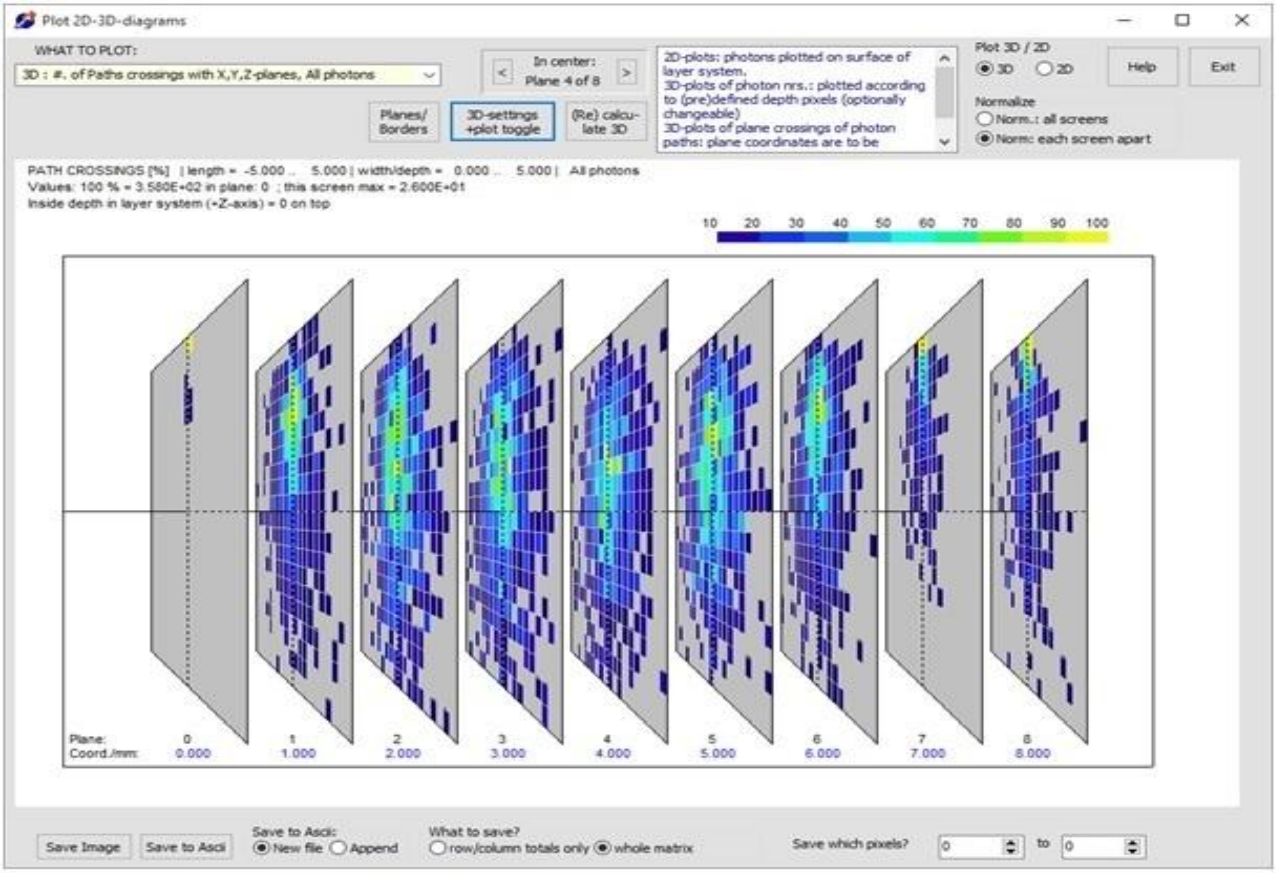


Fig. 16. Path tracking of Fig. 15: crossings with predefined planes; 1 layer.

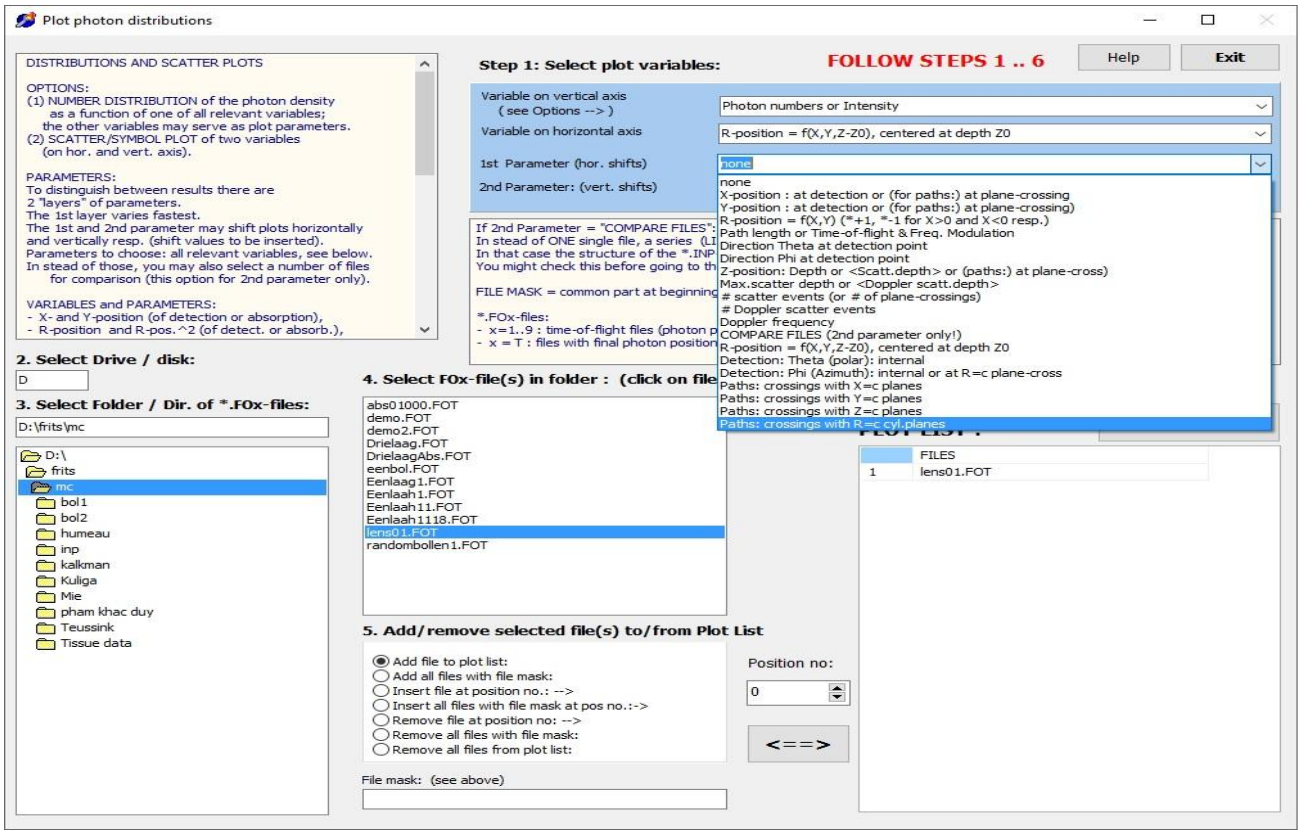


Fig.27. Plot options: choice of axes ("intensity" if vert.axis = none)

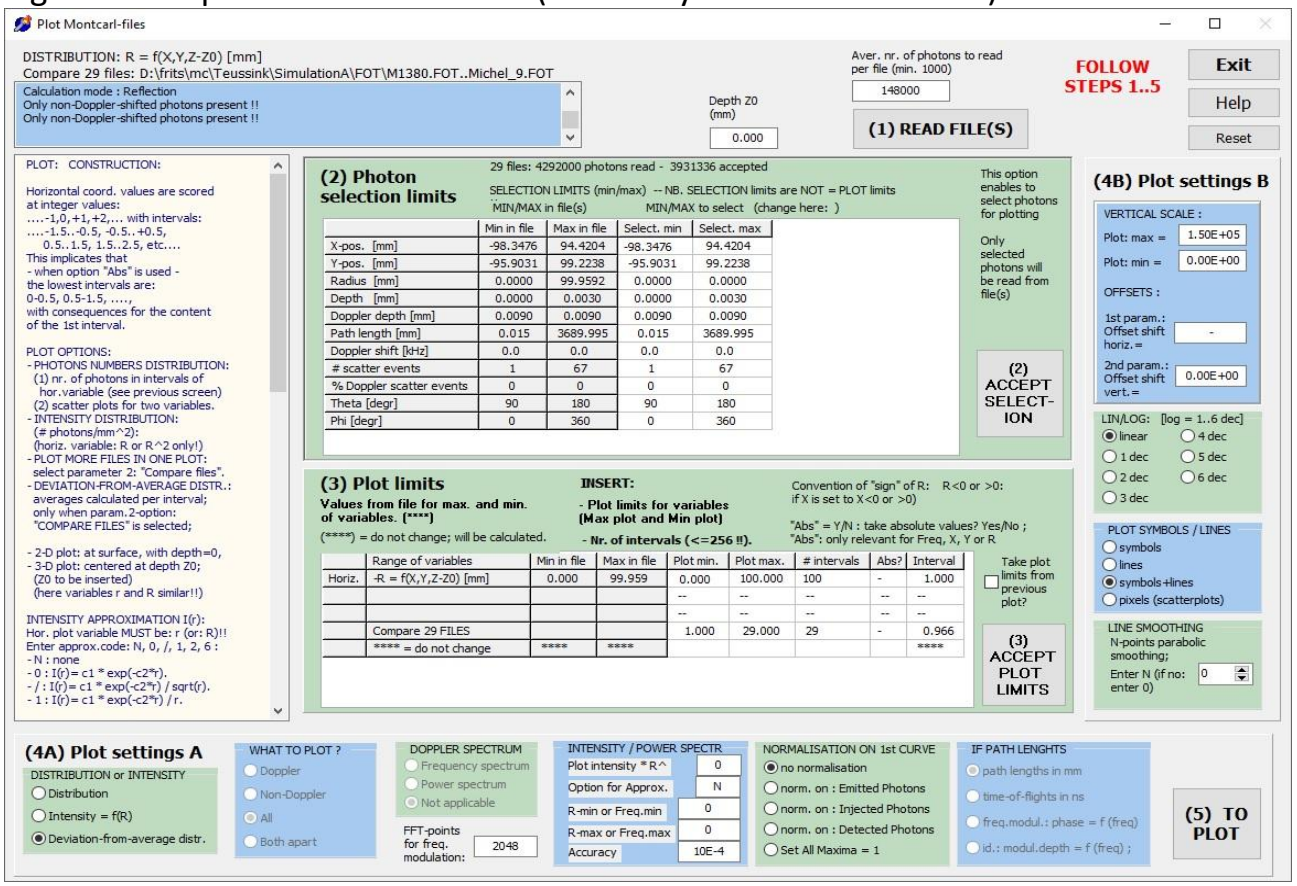


Fig.28. Plot options: settings for an intensity plot

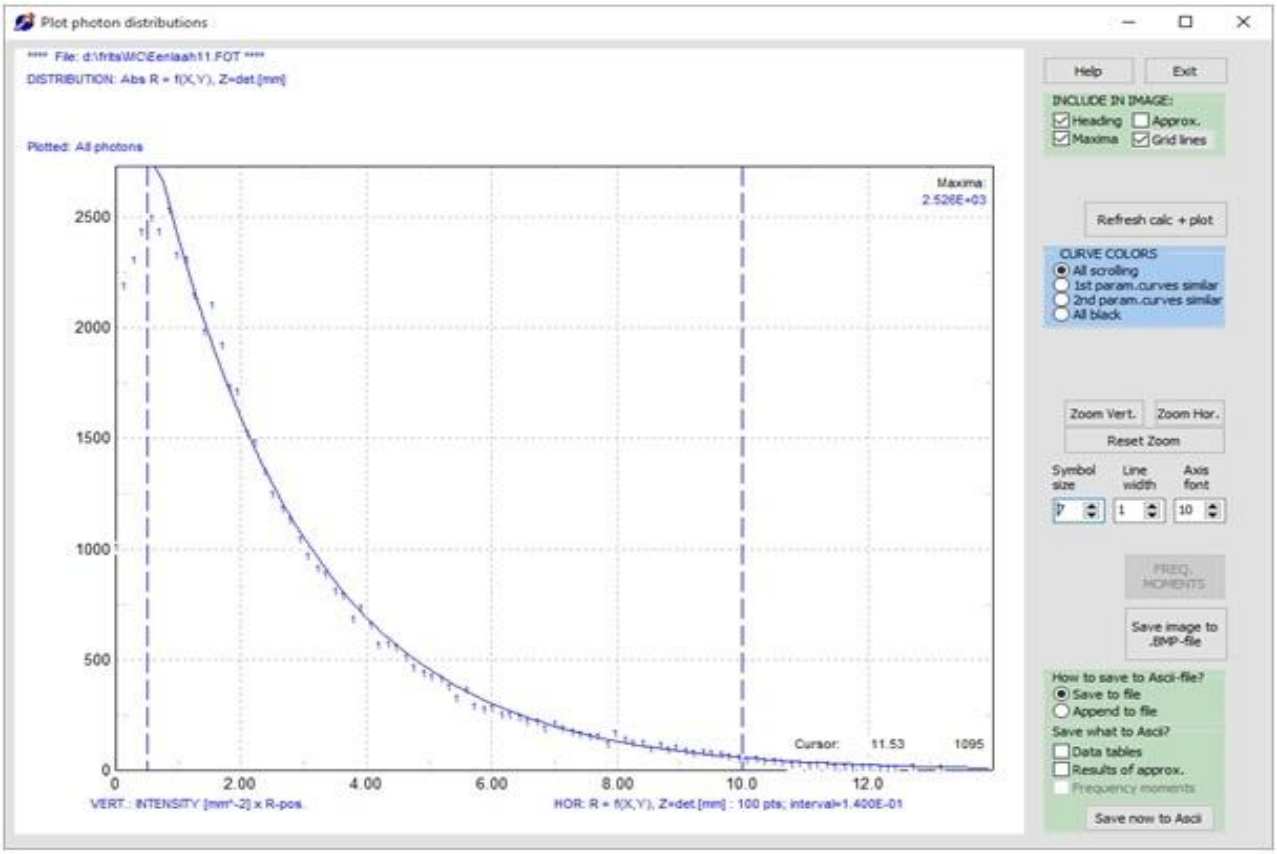


Fig. 29. Intensity plot, with model fitting. Plots of >1 runs optional (with shifts).

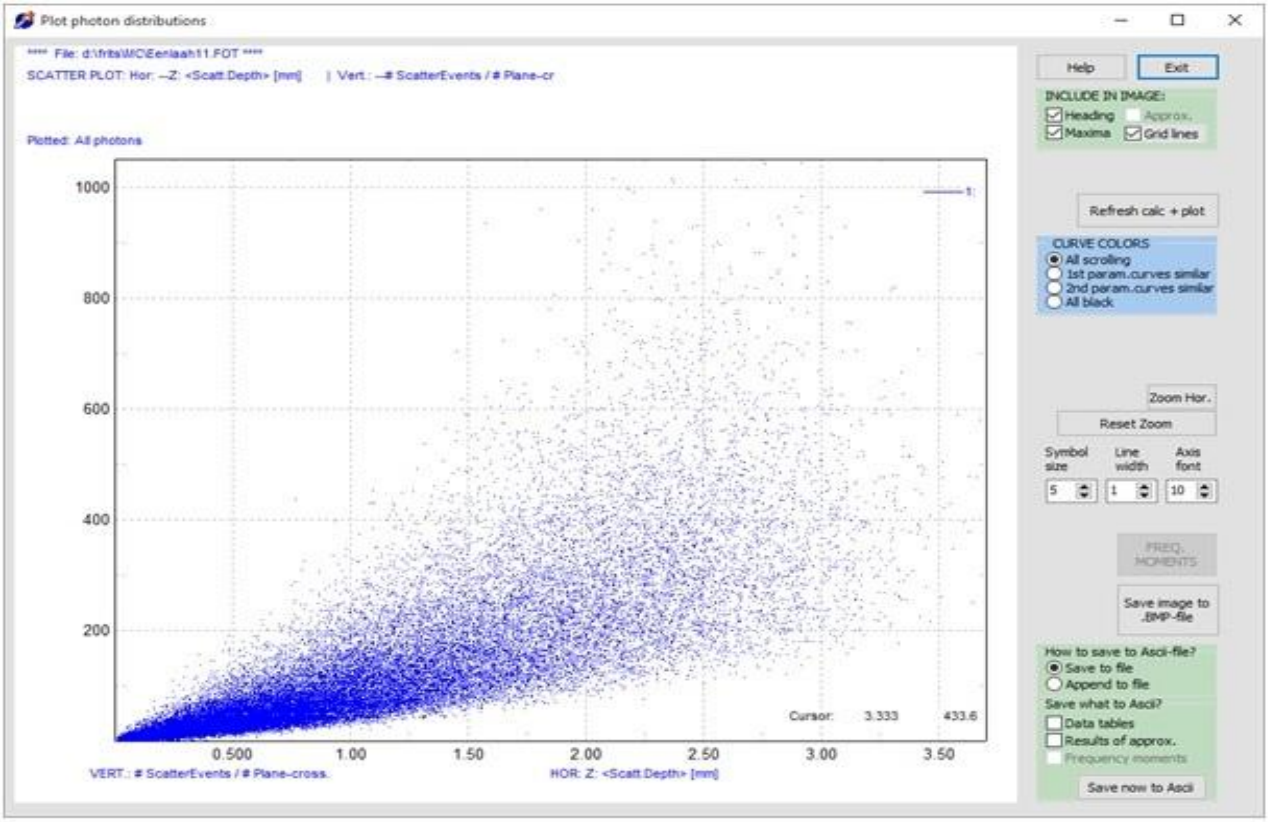


Fig. 30. Scatter plot of results. Plots of >1 runs optional (with vert./hor. shifts)

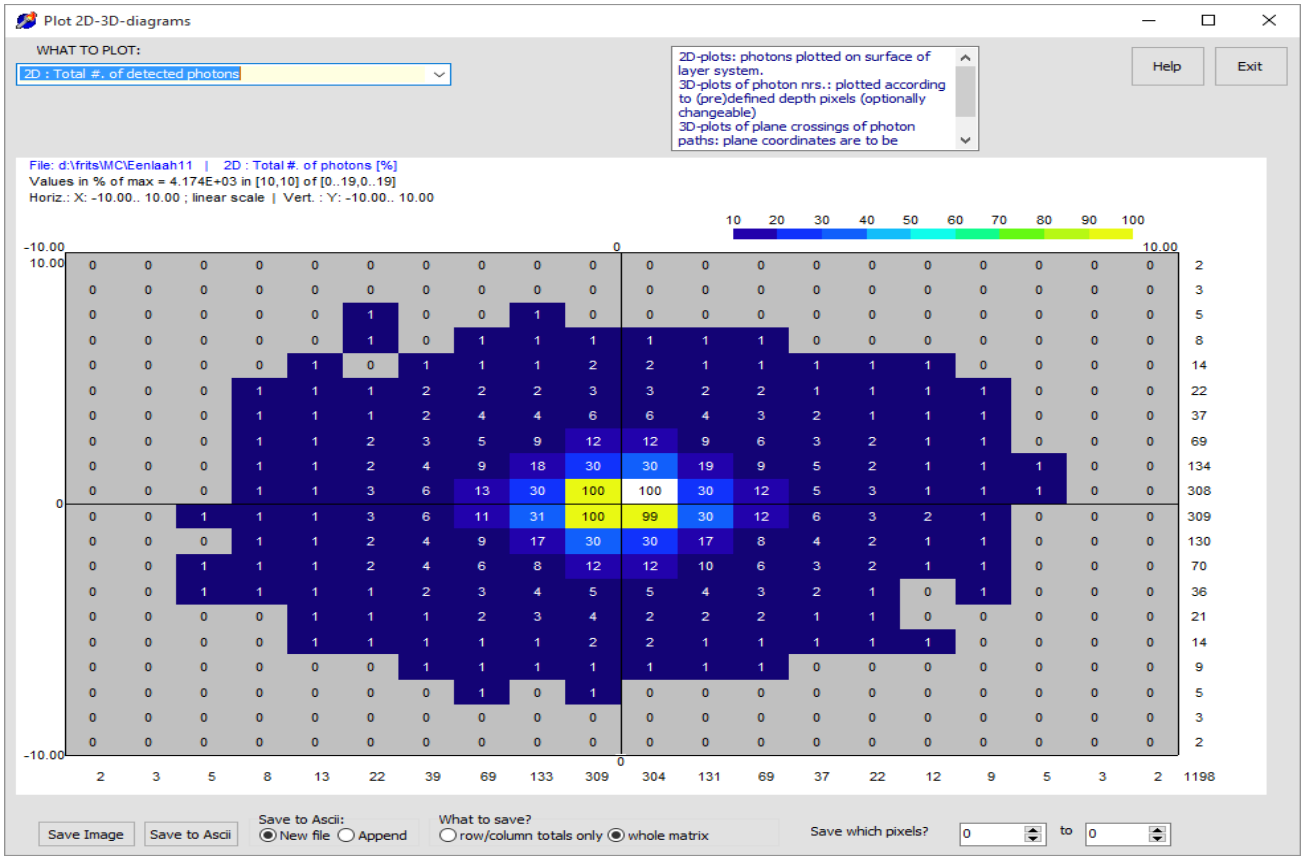


Fig. 31. 2D-plot of results: total number of re-emerging photons.

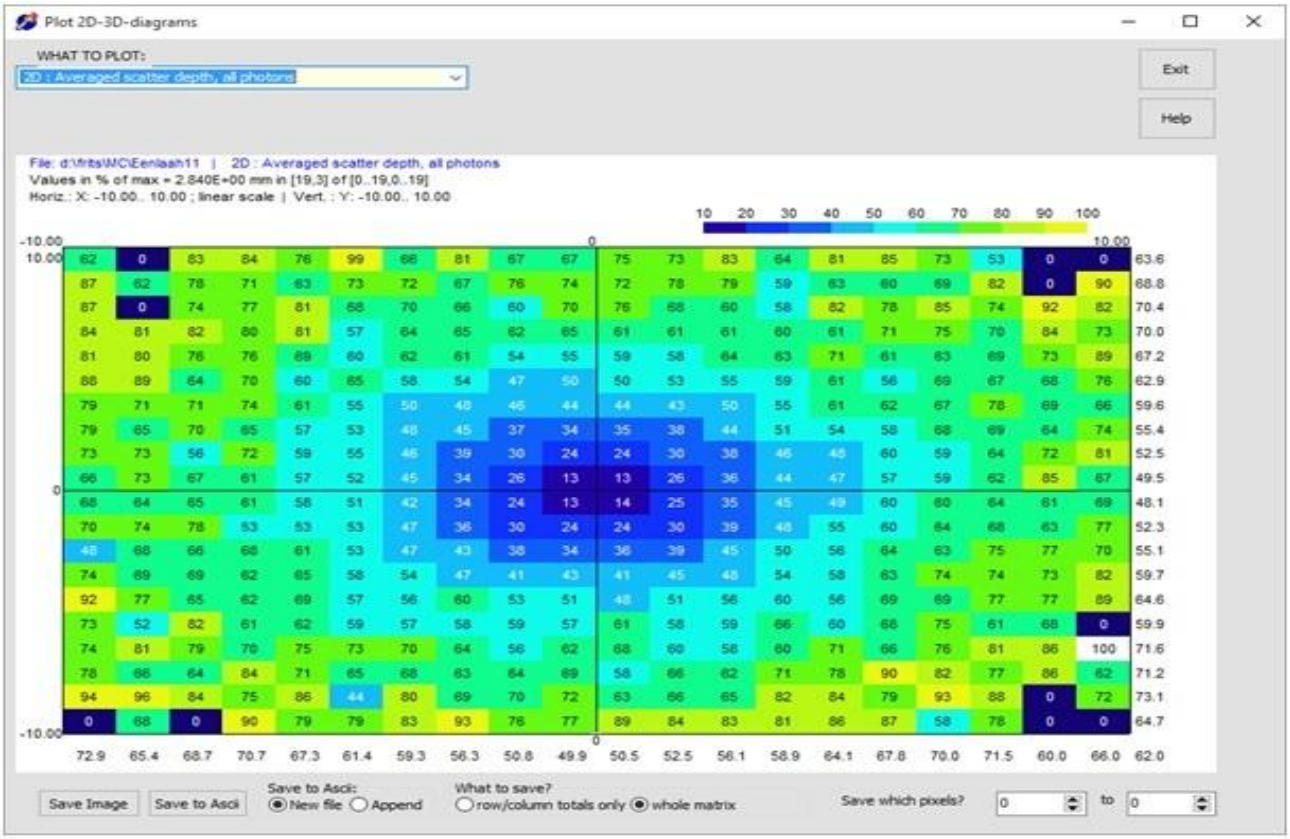


Fig.32. 2D-plot of results: average scattering depth of all photons.

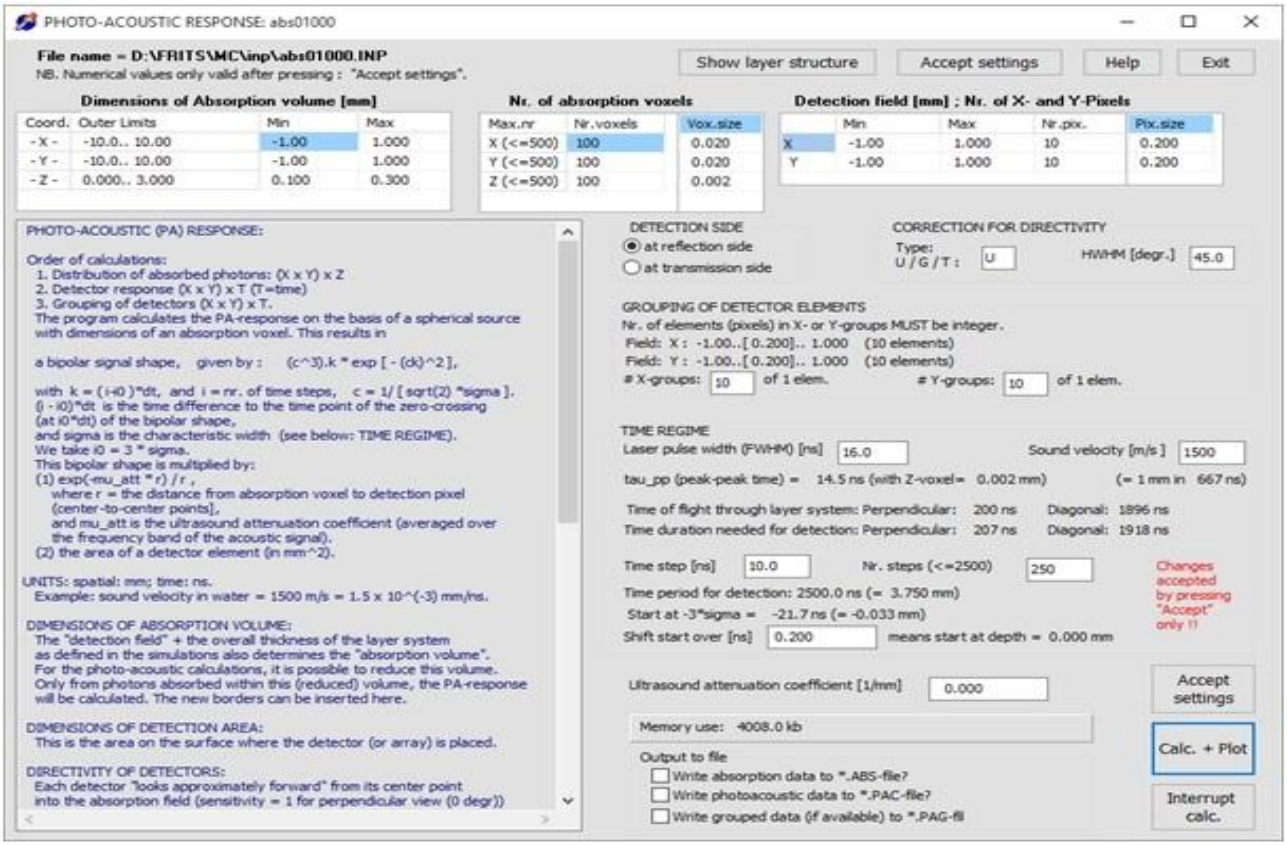


Fig. 33. Photo-acoustic response of absorbed photons: settings

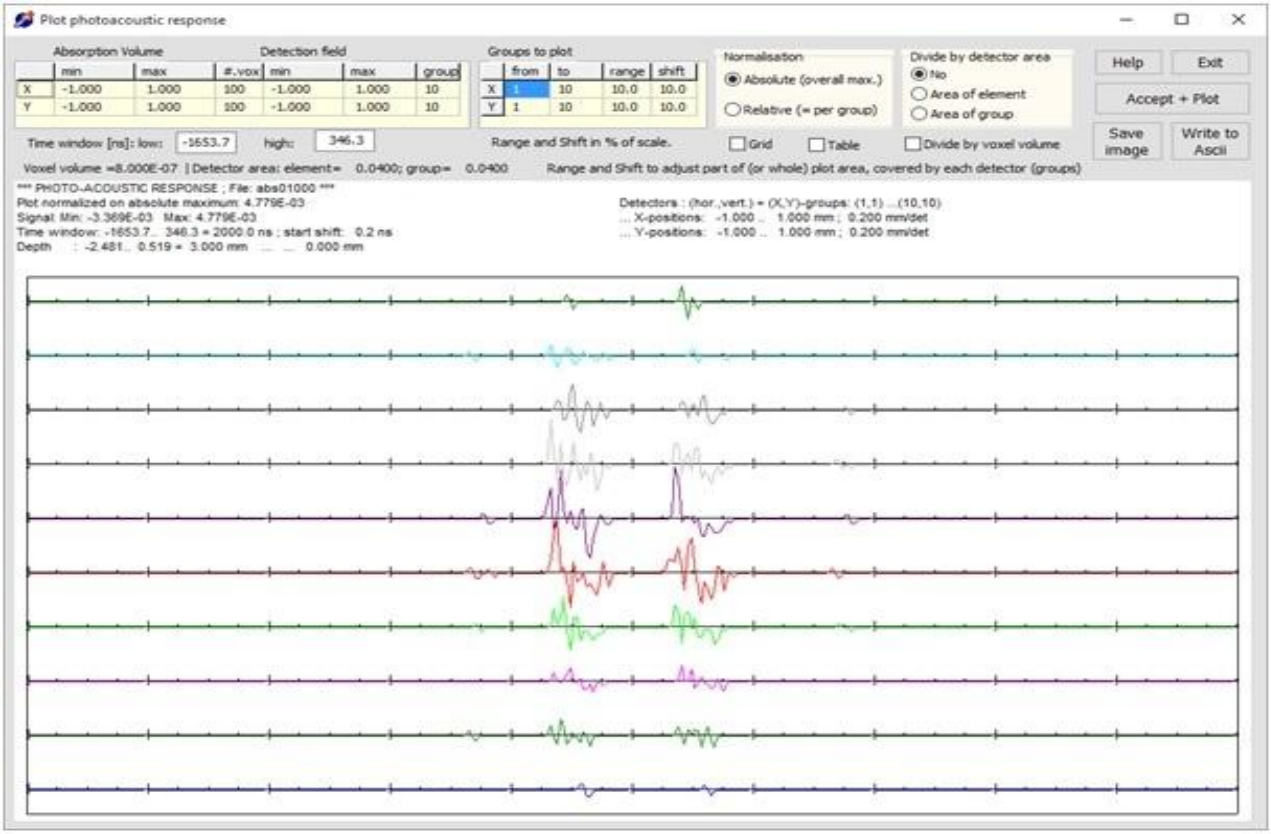


Fig. 34. Photo-acoustic response in 10x10–detector array of 1 tube (Fig.21)

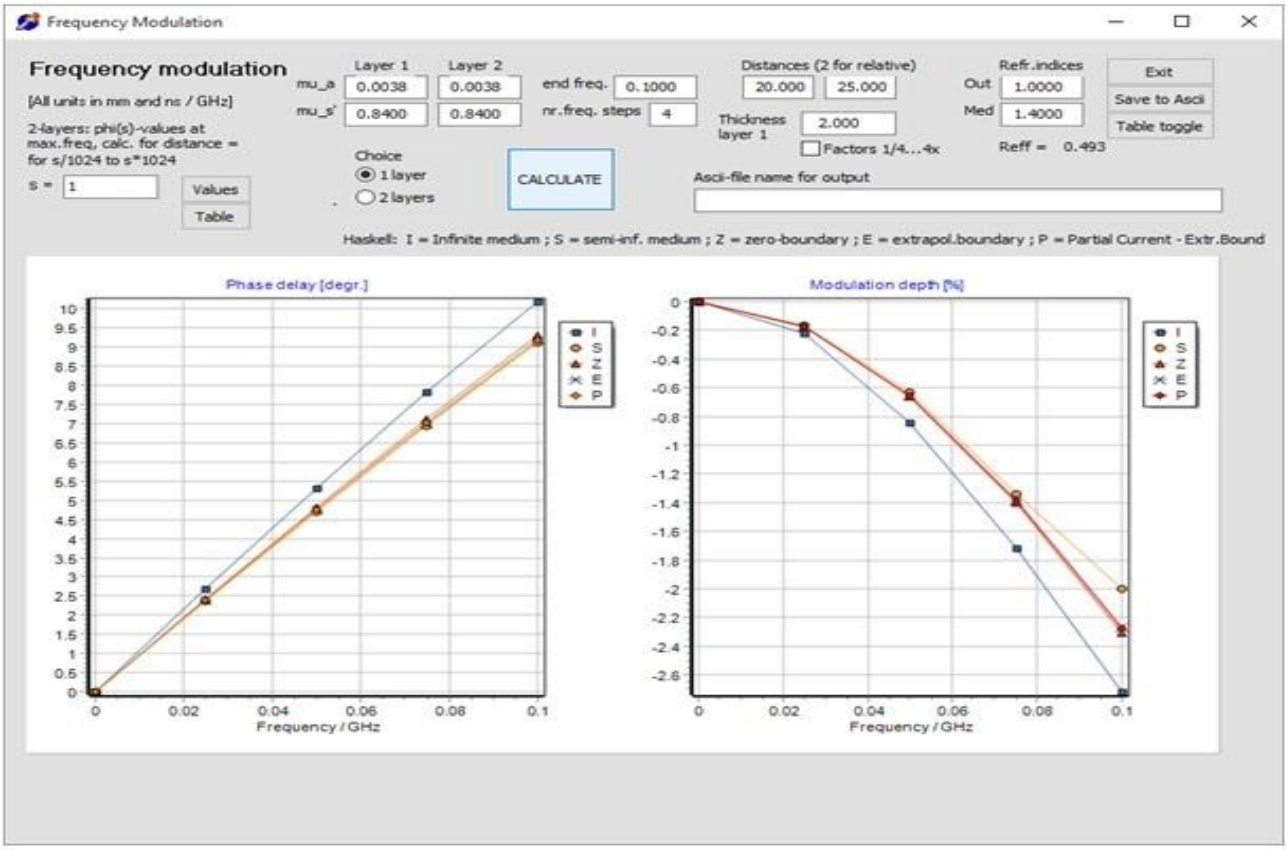


Fig.35. Extra: frequency modulation of GHz-signals in tissue layers.

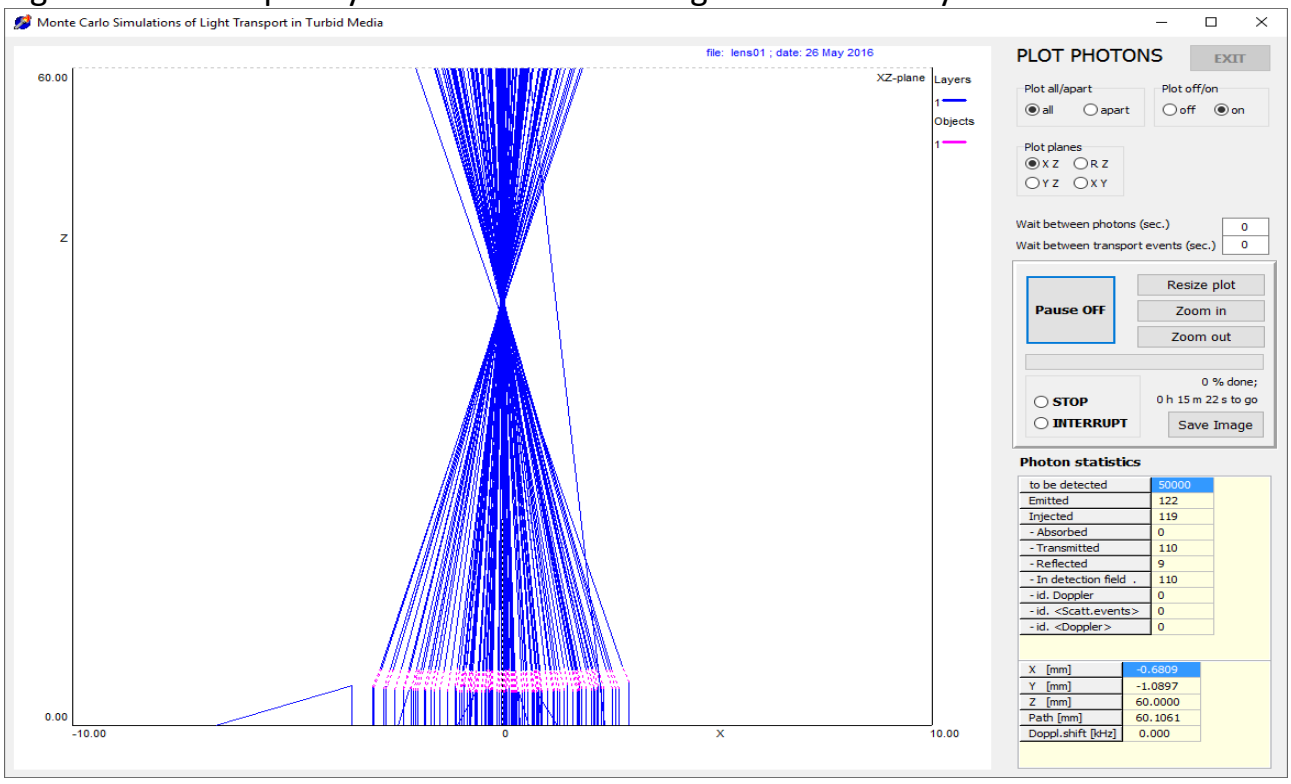


Fig.36. Imaging through a thick convex-concave lens with a few scatterers

---