Joint ICTP-IAEA Workshop on Monte Carlo Radiation Transport and Associated Data Needs for Medical Applications
28 October – 8 November 2024
ICTP, Trieste, Italy

Lecture 17

DOSXYZnrc dose calculations in a phantom

Blake Walters

Metrology Research Centre
National Research Council Canada

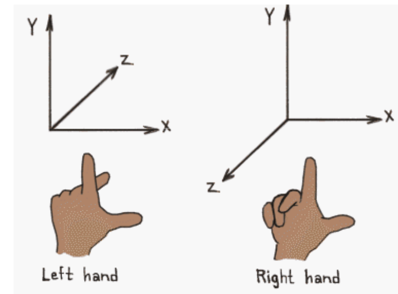
Government of Canada
Gouvernement du Canada
Carleton University
Université de Montréal
nova scotia health

DOSXYZnrc History

- DOSXYZ, for use with EGS4, coded by DWOR in 1987 to show that specialized coding of HOWFAR for rectilinear voxels faster than using generalized macros
- Became basis for a MC timing benchmark in 1992
- Released with BEAM as part of the OMEGA project in 1995
- Updated to incorporate new physics in EGSnrc in 2001: DOSXYZ $\rightarrow$ DOSXYZnrc
- Became an EGSnrc application in mid 00's
- Remains a workhorse for efficient dose calculations in phantoms with rectilinear voxels, including CT-based phantoms

DOSXYZnrc

- Geometry is a rectilinear volume with X-Y plane on the page, X to the right, Y down the page, and the Z-axis into the page (right-handed coordinate system). Voxels can have variable dimensions and material composition.

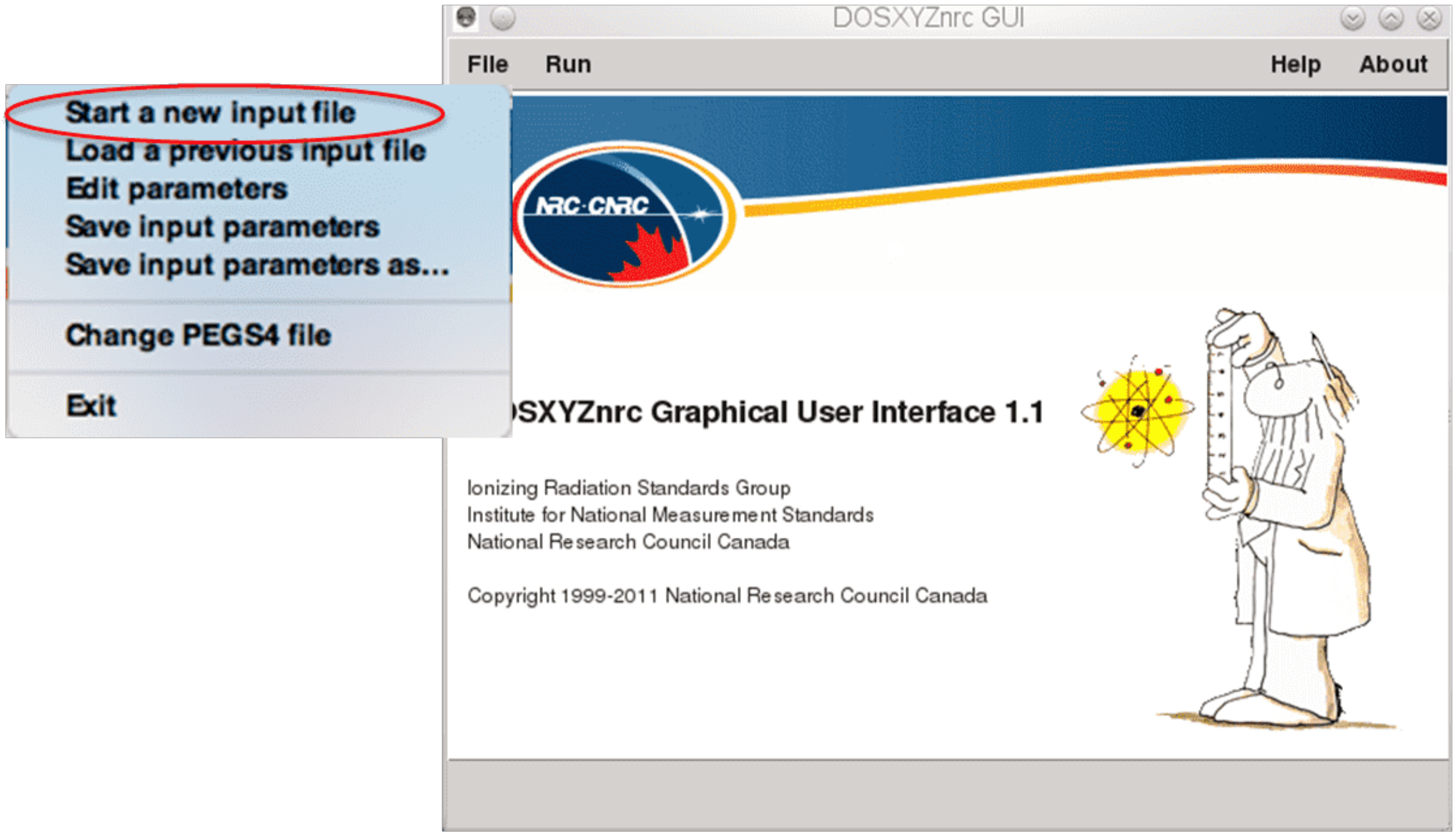


- Dose components based on where particles have passed through or interacted in, or whether the particle is a primary or secondary particle, etc.

- Uses several calculation-efficiency-increasing techniques for fast dose calculations

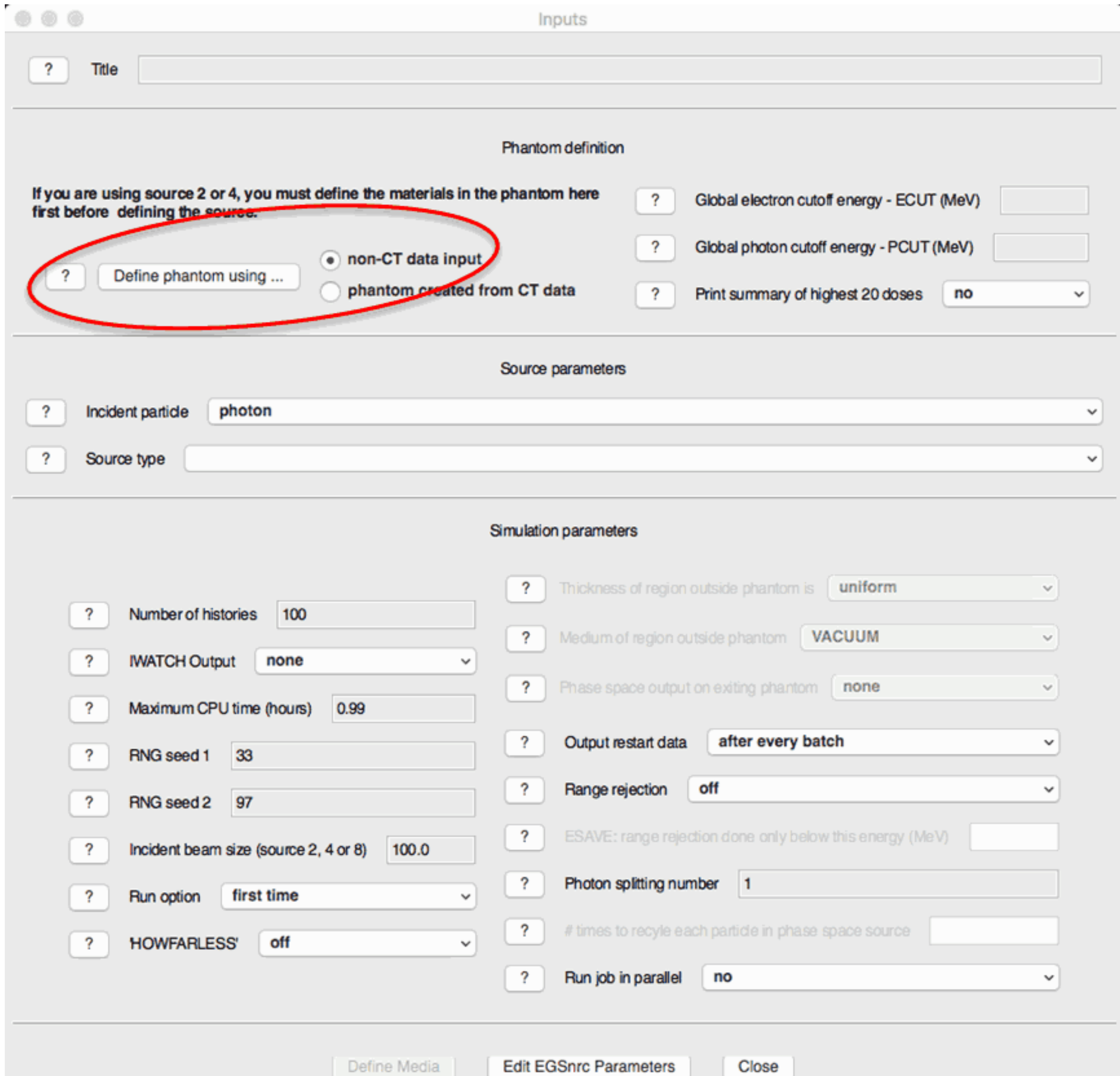
GUI

- type "`dosxyz_gui`" or "`dosxyznrc_gui`"



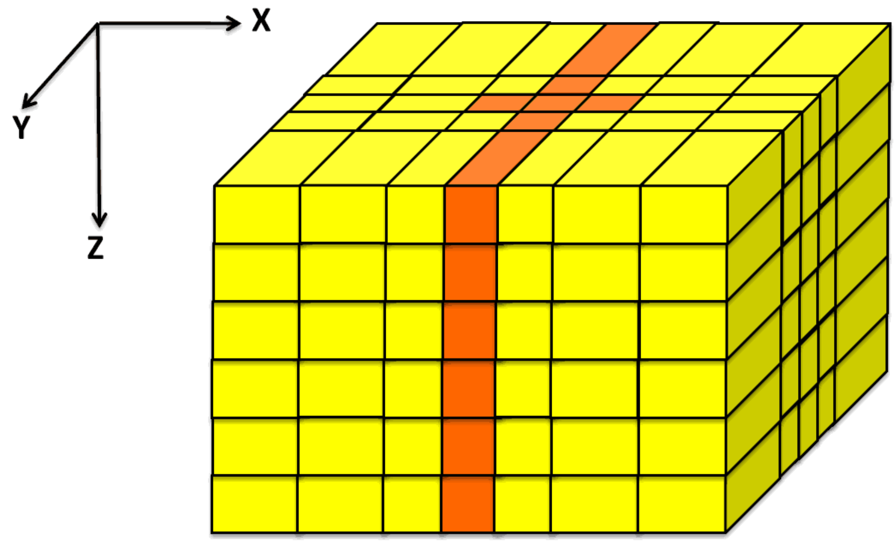
- option to open with: "`dosxyz_gui inputfile pegsfile`"

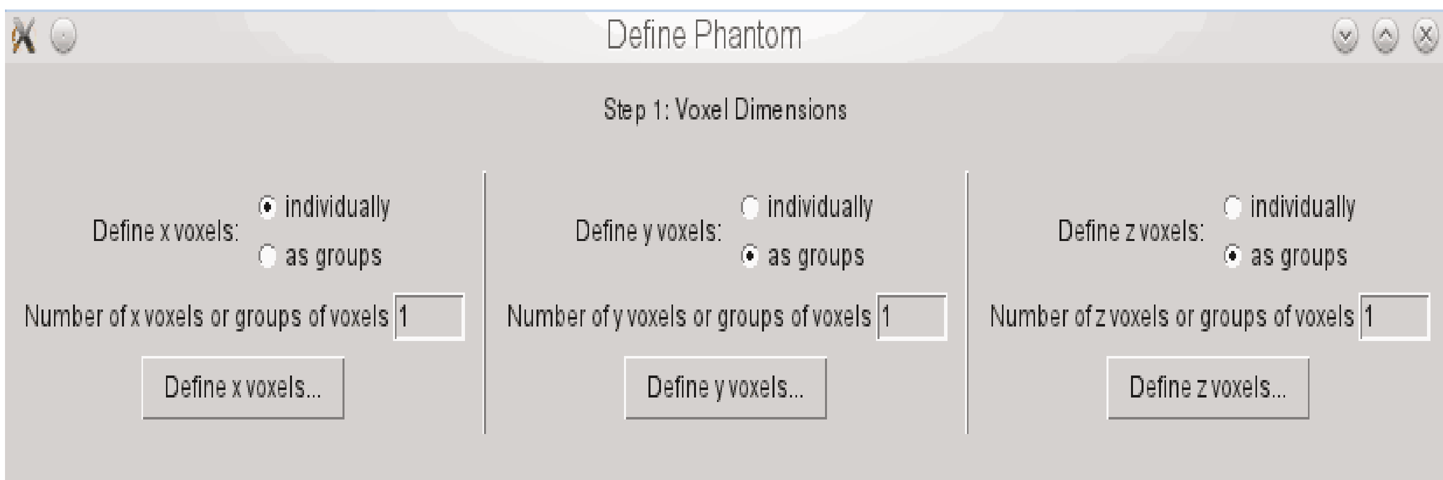
Defining geometry



Phantom definition: non-CT input

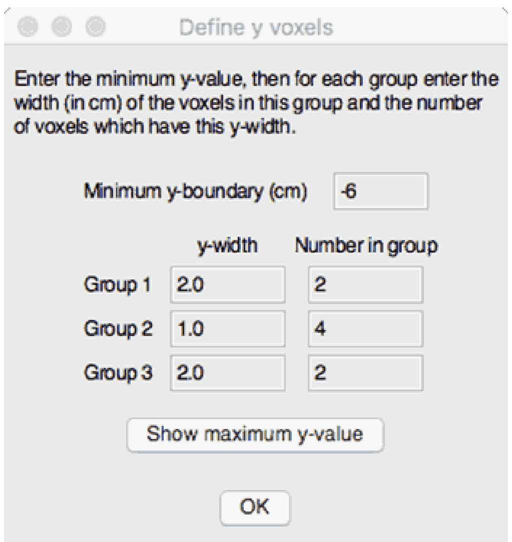
- Individual voxel definition
- Voxel definition by group



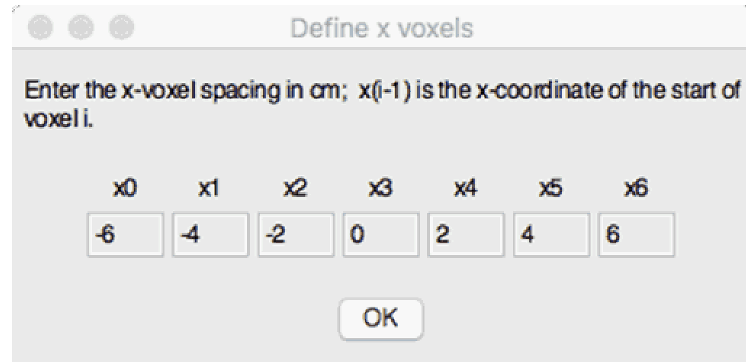


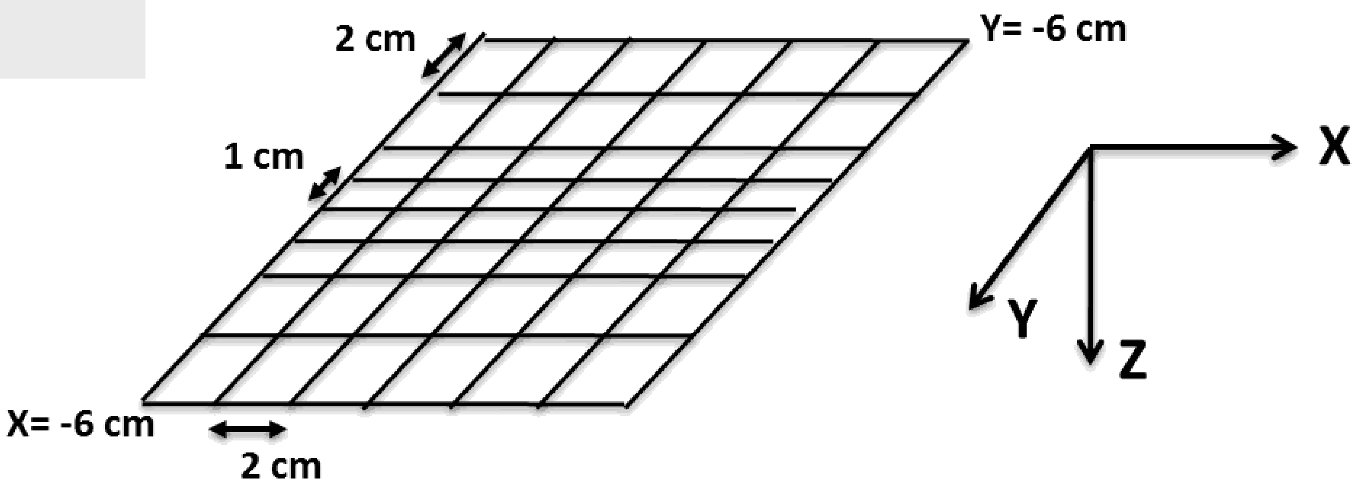
Phantom definition: individual vs groups

group

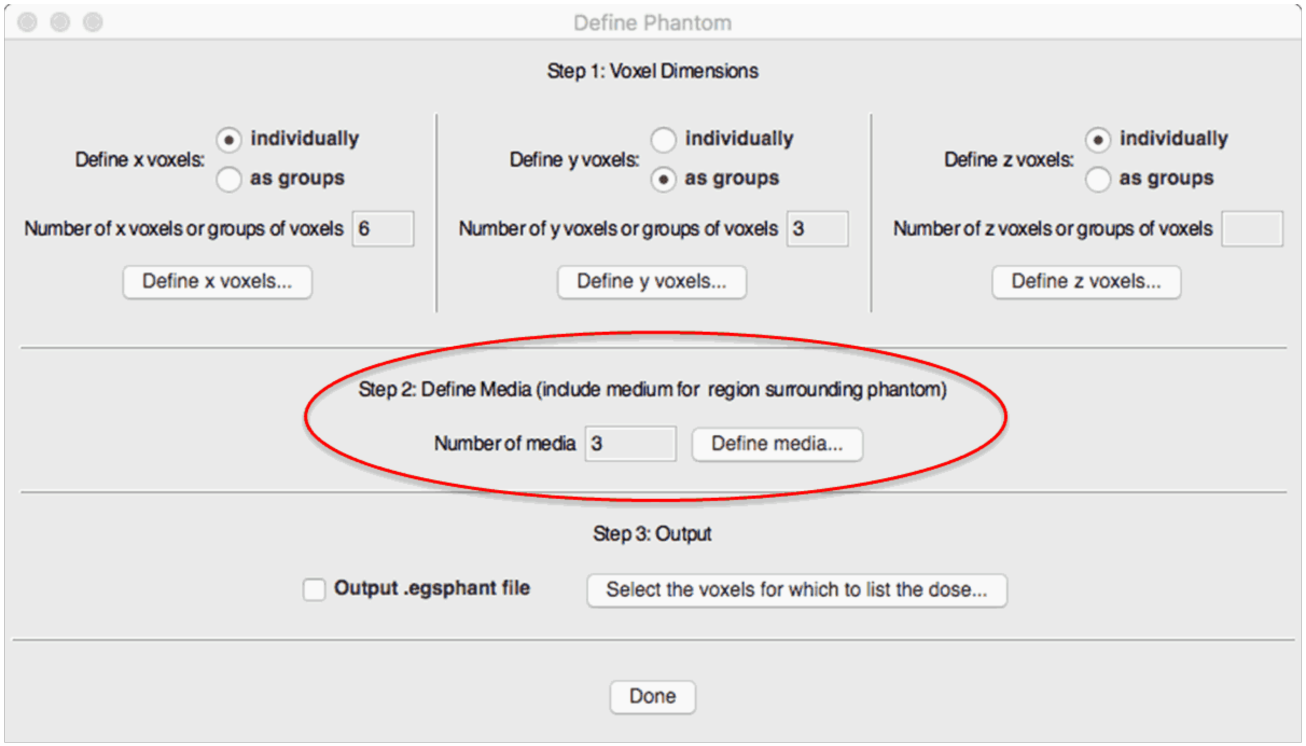


individual



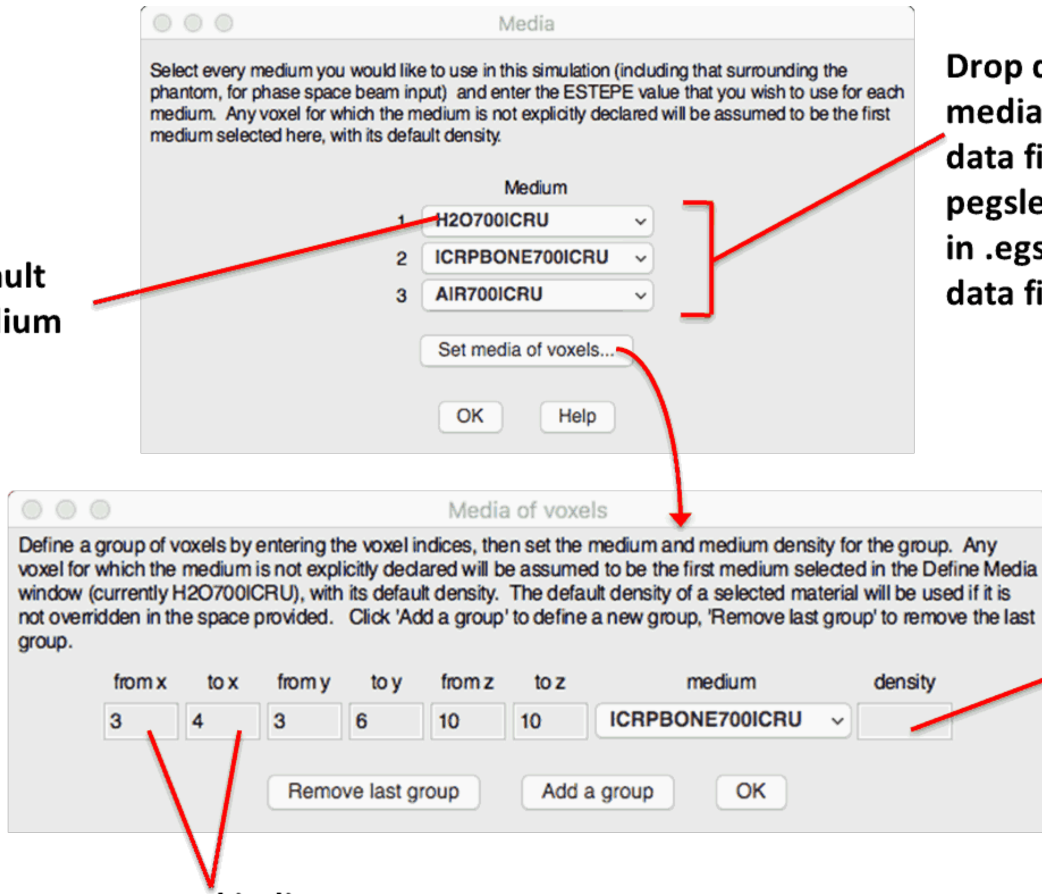


Phantom media definition: non-CT input



- **input no. of different media for entire simulation (including surrounding air)**

Phantom media definition (cont.)



Default medium

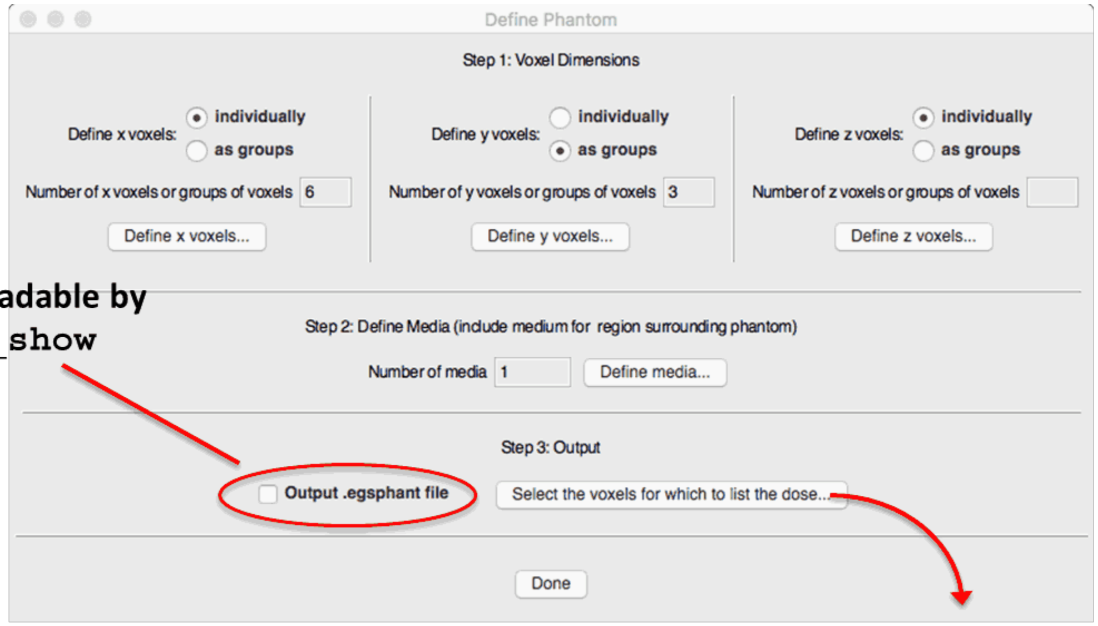
Drop down menus list all media available in pegs data file or, if running pegsless, all media defined in .egsinp file and material data file

leave blank to use density in pegs(less) definition

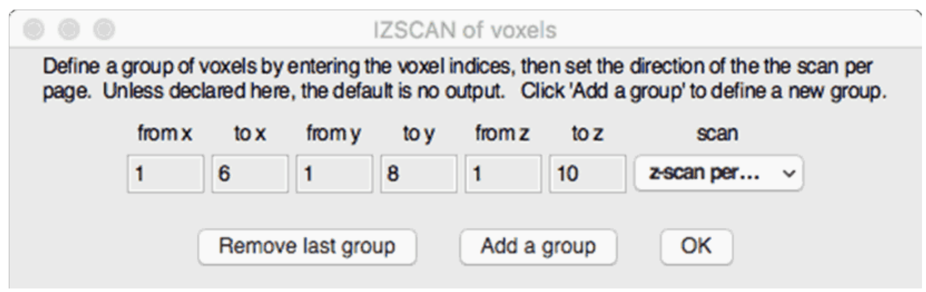
voxel indices

Phantom output control

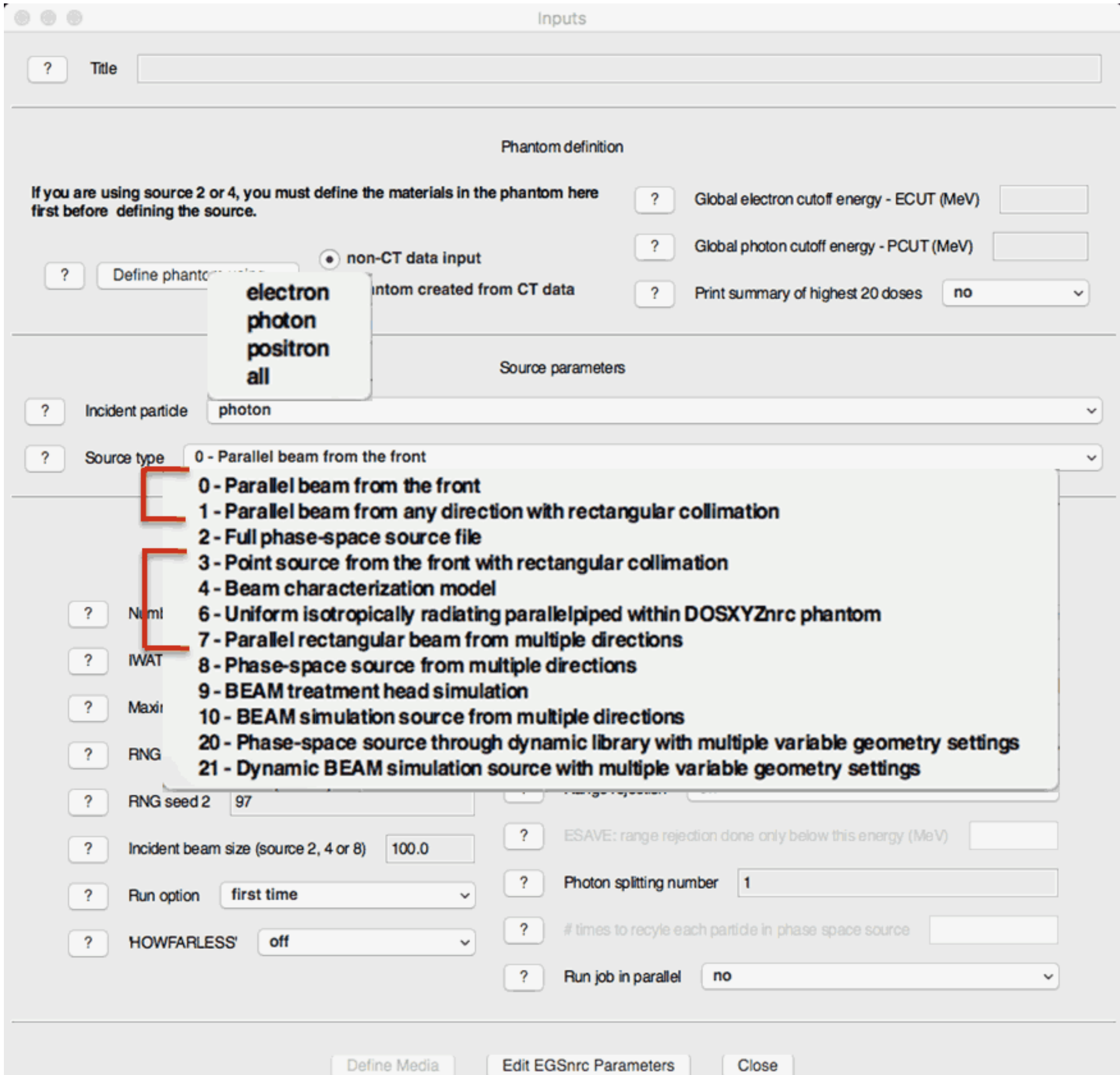
format readable by
`dosxyz_show`



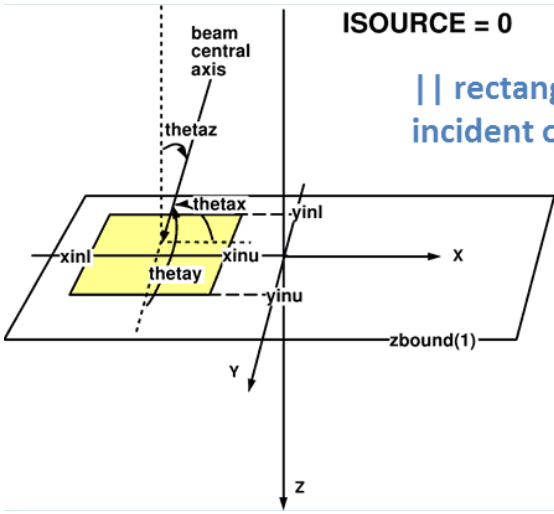
- controls output to .egslst file
- might want to have no output for phantom with many voxels



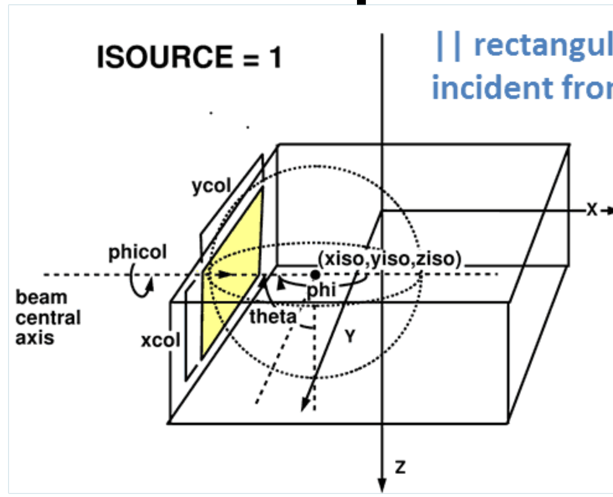
Built-in sources



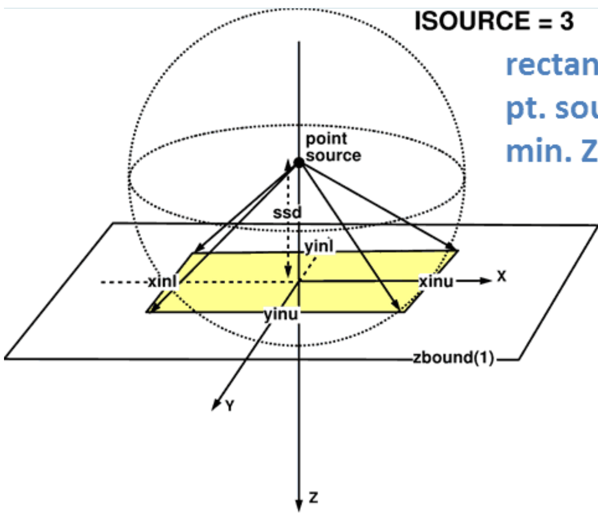
Built-in sources: examples



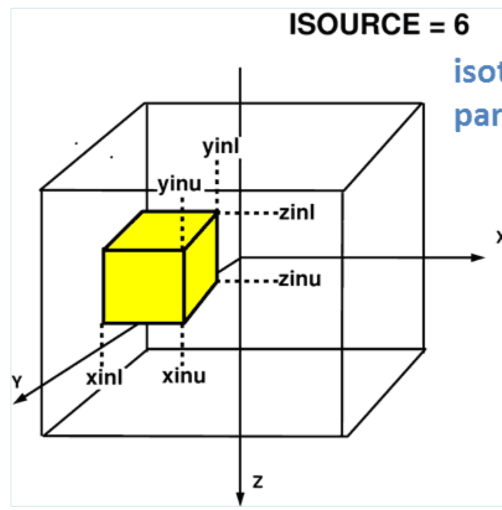
| | rectangular beam incident on min. Z



| | rectangular beam incident from any direction



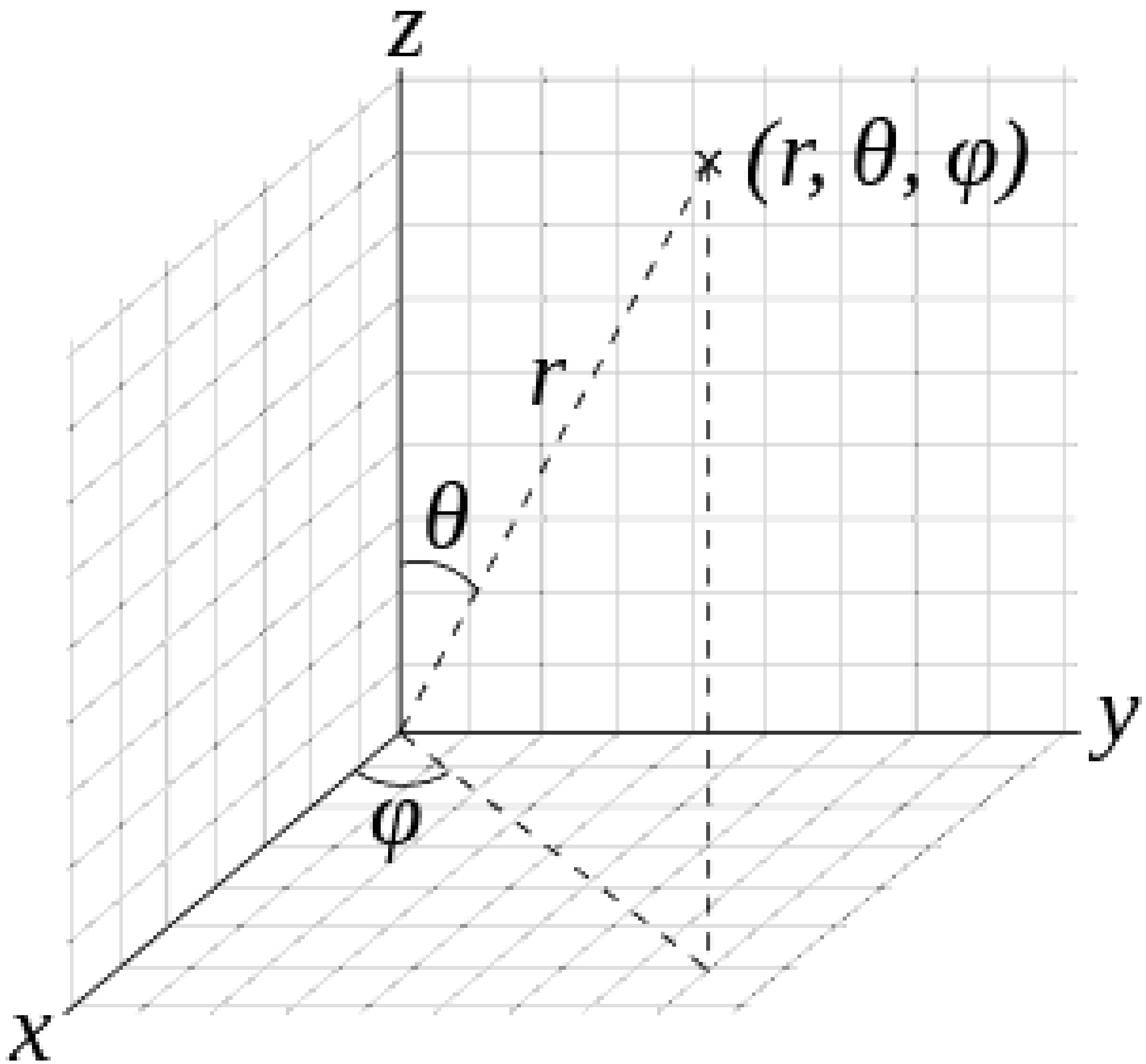
rectangular collimated pt. source incident on min. Z



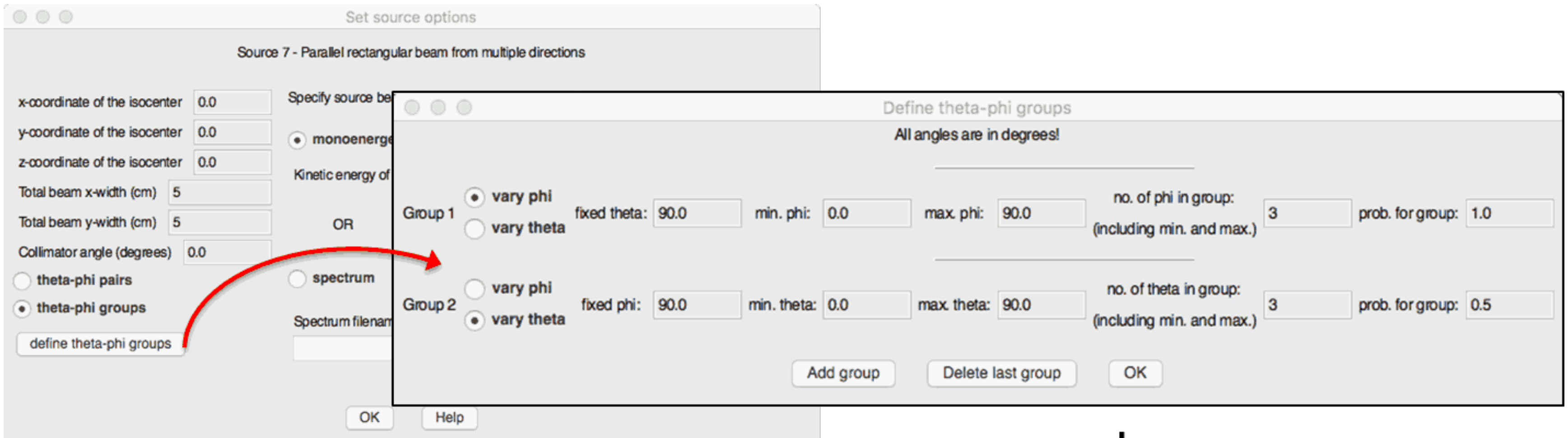
isotropically radiating parallelepiped

See DOSXYZnrc manual (PIRS-794) for more details on all these sources!

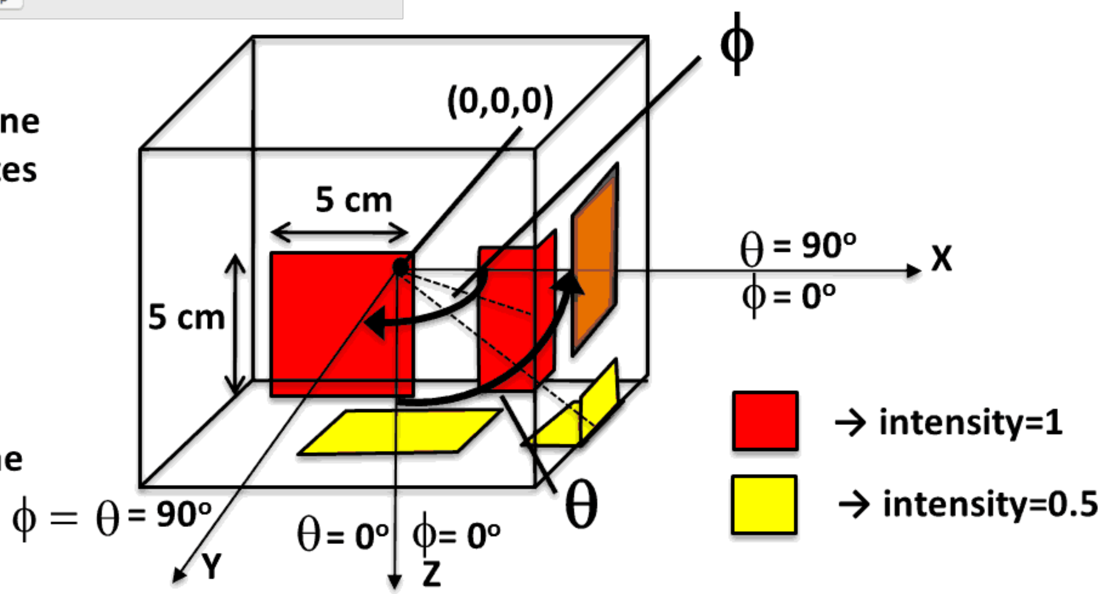
Polar coordinate system



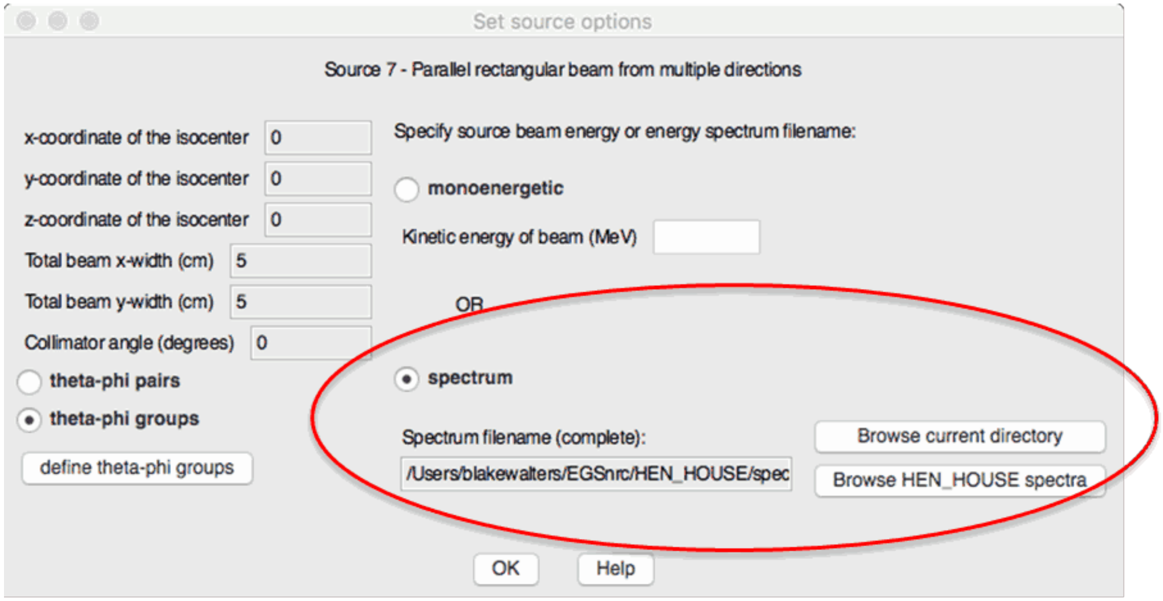
isource=7: parallel beam incident from multiple angles



- isocentre, theta, phi define points in polar coordinates
- theta = angle wrt +Z axis
- phi = angle wrt +X axis
- collimator angle (phicol) rotates the source clockwise in its own plane



Built-in sources (energy spectra)



- many sample spectra contained in `$HEN_HOUSE/spectra`

spectrum file format:

```
 60-Co  spectrum Rogers et al 87
27, 0.01,0
0.05, .0000362
0.10, .00132
0.15, .01301
0.20, .02561
0.25, .03763
0.30, .03554
```

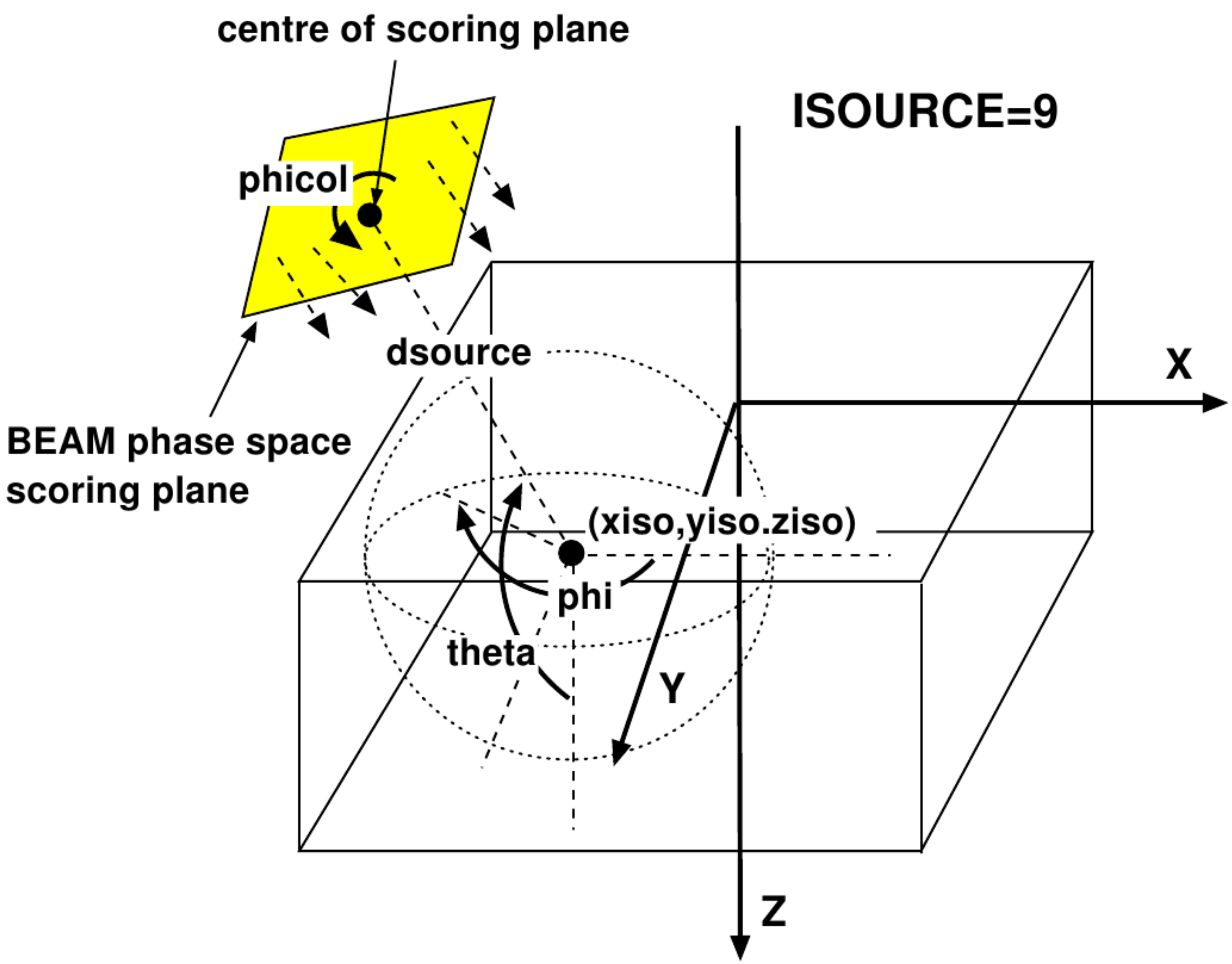
title (80 char)

nbin, Emin , mode (0=cts/bin, 1=cts/MeV)

Emax(1), prob(1)

Emax(2), prob(2)

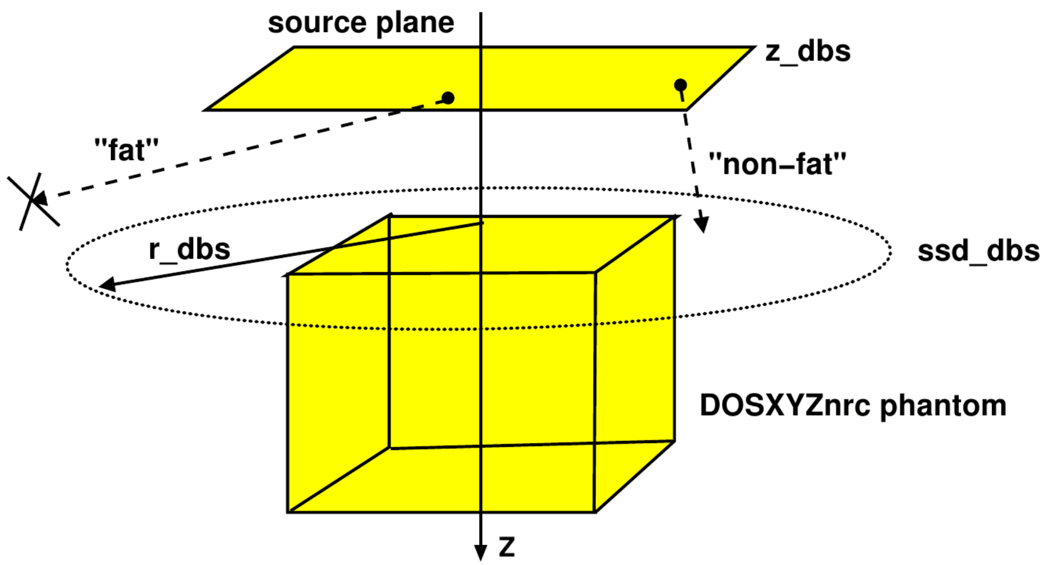
Phase space and BEAM Sources



Other Inputs for Phase Space Sources

Rejecting “fat” photons if DBS used to generate phase space source

- “fat” particles will compromise statistics
- `i_dbs`=1
- `r_dbs`=DBS splitting radius
- `ssd_dbs`=SSD at which splitting radius was defined
- `z_dbs`=Z (in accelerator frame) where phase space file was scored



Other Inputs for Phase Space Sources (cont.)

splitting charged particles (e_split)

- Only used with photon splitting n_split $>$ 1
- Charged particles split e_split times (with weight reduced by 1/e_split) as soon as they enter the phantom geometry
- Prevents high-weight contaminant e-'s from compromising dose statistics
- For maximum efficiency, set e_split=n_split

Reduce uncertainty by particle redistribution (ISMOOTH)

- On re-use, each particle's (X,Y) and (U,V) are shifted to:
  1. (-X,Y) with (-U,V)
  2. (X,-Y) with (U,-V)
  3. (-X,-Y) with (-U,-V)

Accurate as long as the simulated linear accelerator geometry is symmetric, and the treatment field is centred on the beam axis!

Other Inputs for Phase Space Sources (cont.)

Number of times to recycle each particle before moving on to the next one (NRCYCL)

- avoids restarts $\Rightarrow$ **essential for accurate statisitics if no. of histories $>$ no. of particles in phase space source!**

- if set to 0 (and not using only positrons OR filtering incident particles based on LATCH), automatically calculated using:

$$\mathrm{NRCYCL} = \mathrm{INT}[\frac{\text{no. of histories}}{\text{no. of particles with selected charge}}]$$

- if source restarted more than once or more than 50% of source gets resampled in a restart (particles miss geometry, or are rejected because they are multiple passers or outside BEAM_SIZE), recalculate NRCYCL using formula in manual PIRS-794.

Other inputs for BEAM simulation source

Rejecting “fat” photons if DBS used to generate phase space source

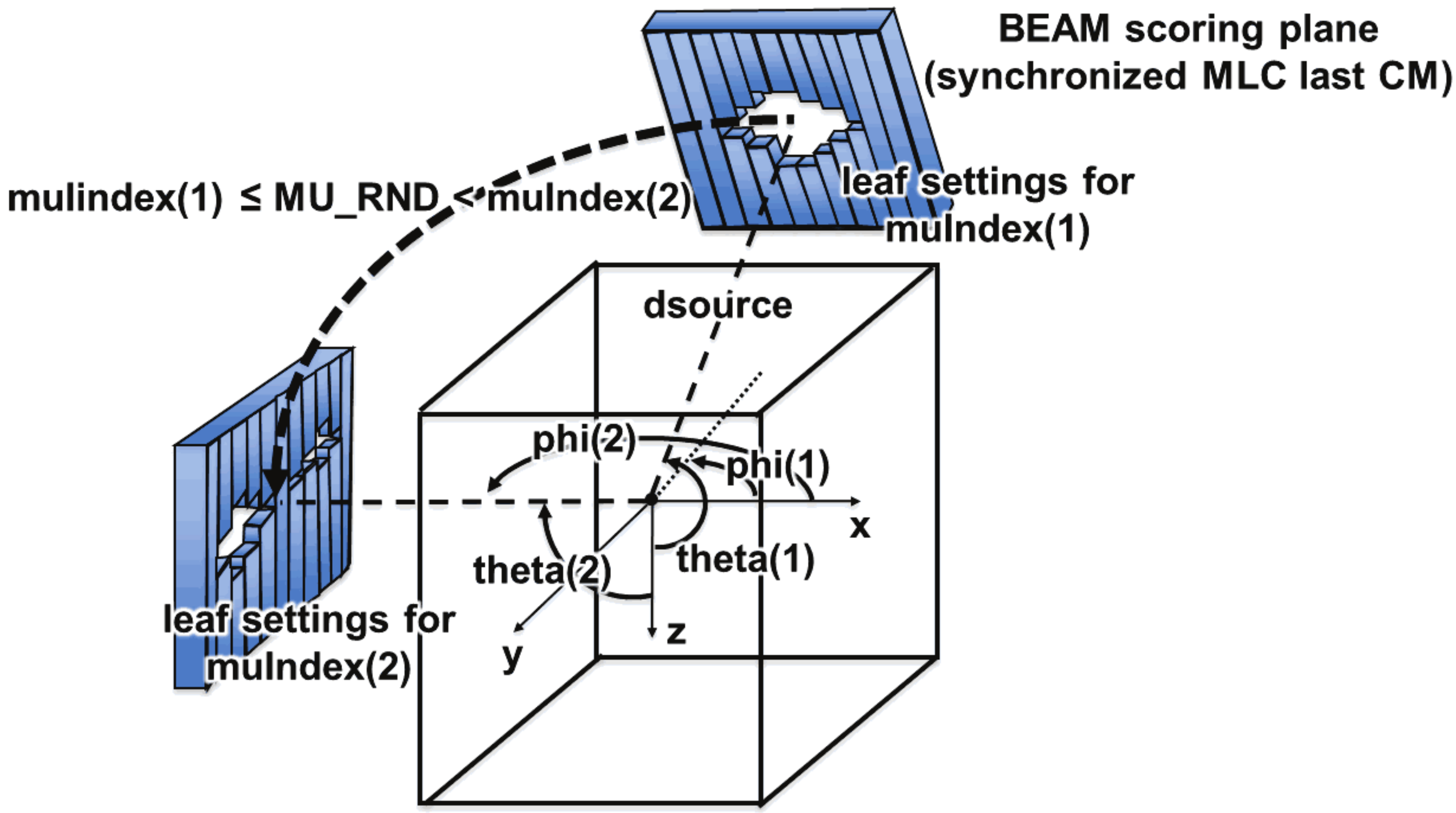
- `i_dbs=1`
- Particle still carries flag indicating if it’s “fat” or not, so this is all we need

Splitting charged particles (`e_split`)

- Same as for phase space source

Synchronized phase space and BEAM sources

- Simulate continuous motion of the source plane between user-defined control points.
- Developed by Tony Popescu and Julio Lobo (PMB 55 (2010) 4431–4443)



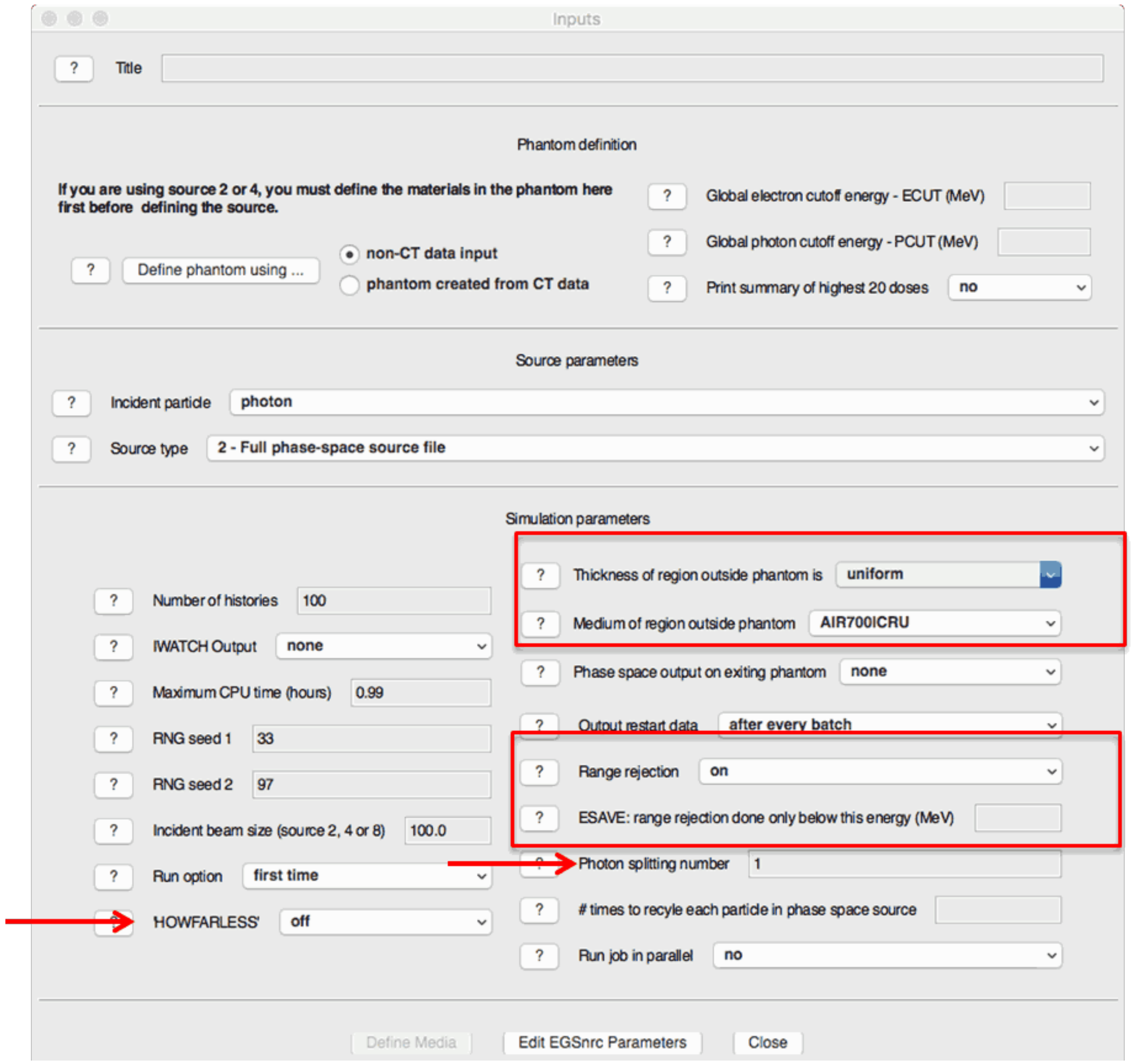
Efficiency Increasing Techniques

- Range rejection (`IREJECT`): Can save 10 % to 17 % on computing time but for smaller voxels it saves less time (3 % to 4 % for 2.5 mm^3 voxels). For phantoms with some large voxels, the savings will be larger, especially using the `dsurround` option.

- Photon splitting (`n_split`): Up to 6.5 $\times$ efficiency increase depending on energy, field size, and voxel size. Potentially *eliminates need to use phase space files*.

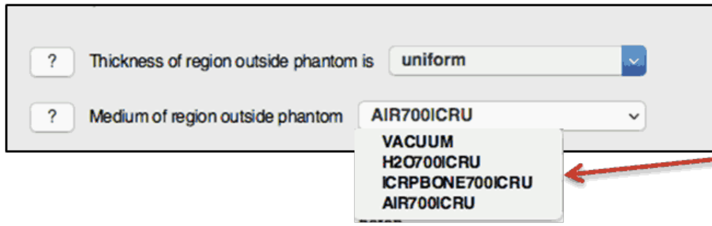
- HOWFARLESS mode (`i_howfarless`): improves efficiency by 50 % – 90 % for BEAMnrc photon beams and by a factor of 3 – 5 for monoenergetic electron beams.

Efficiency Increasing Techniques: GUI



dsurround (region surrounding phantom)

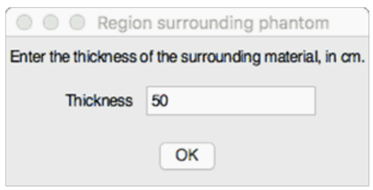
- phase space and BEAMnrc simulation sources only

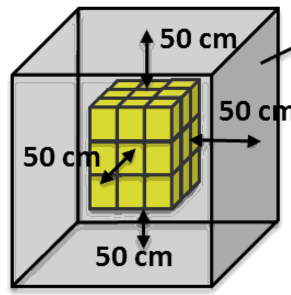


menu contains all media defined with phantom + vacuum

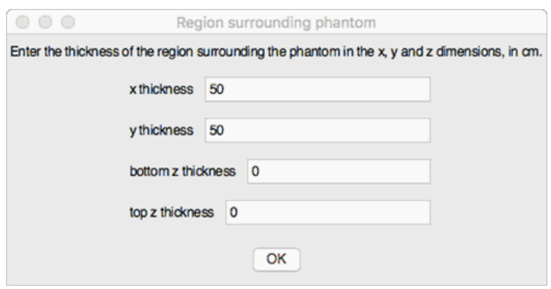
medium surrounding phantom

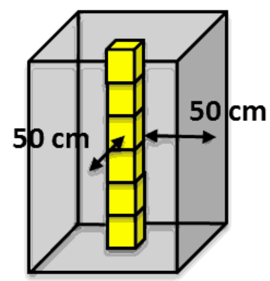
uniform





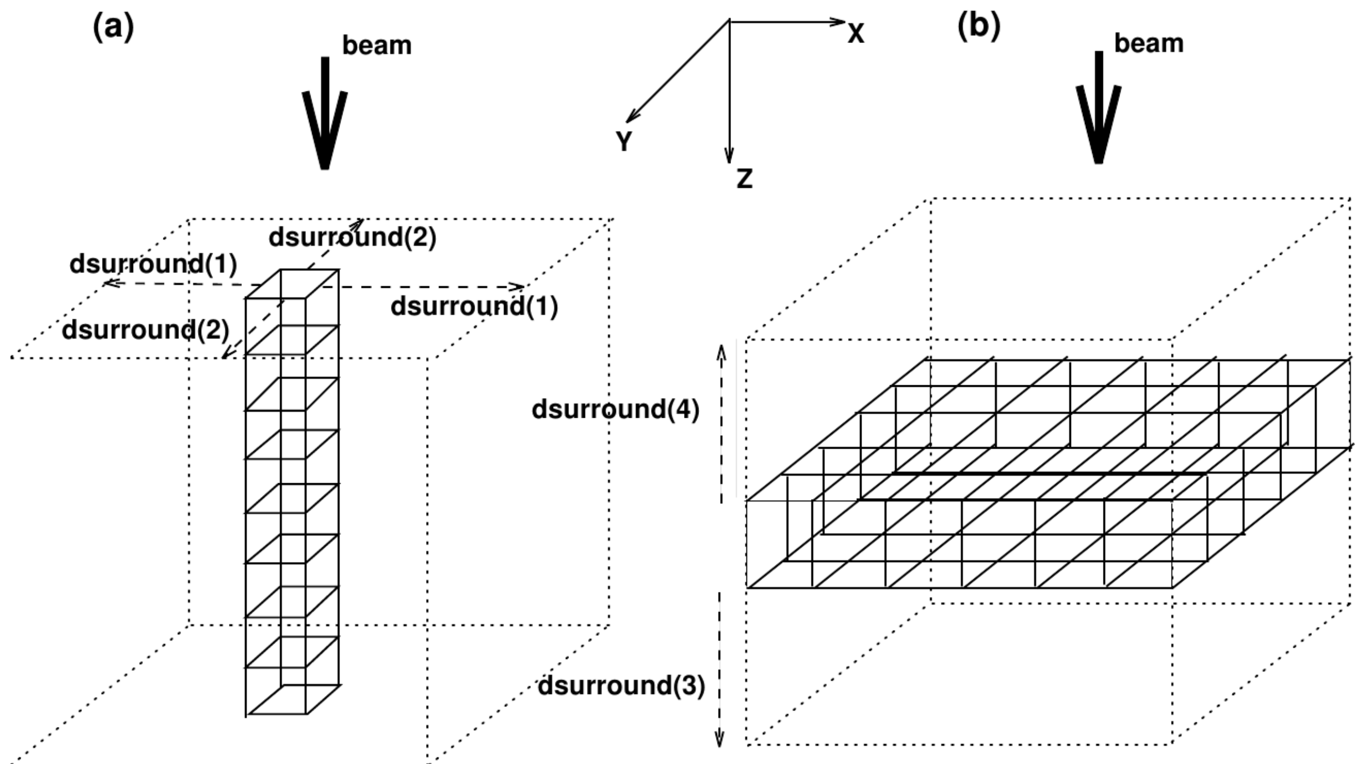
nonuniform





- range rejection can be very efficient

Using non-uniform region surrounding phantom (dsurround,dflag)



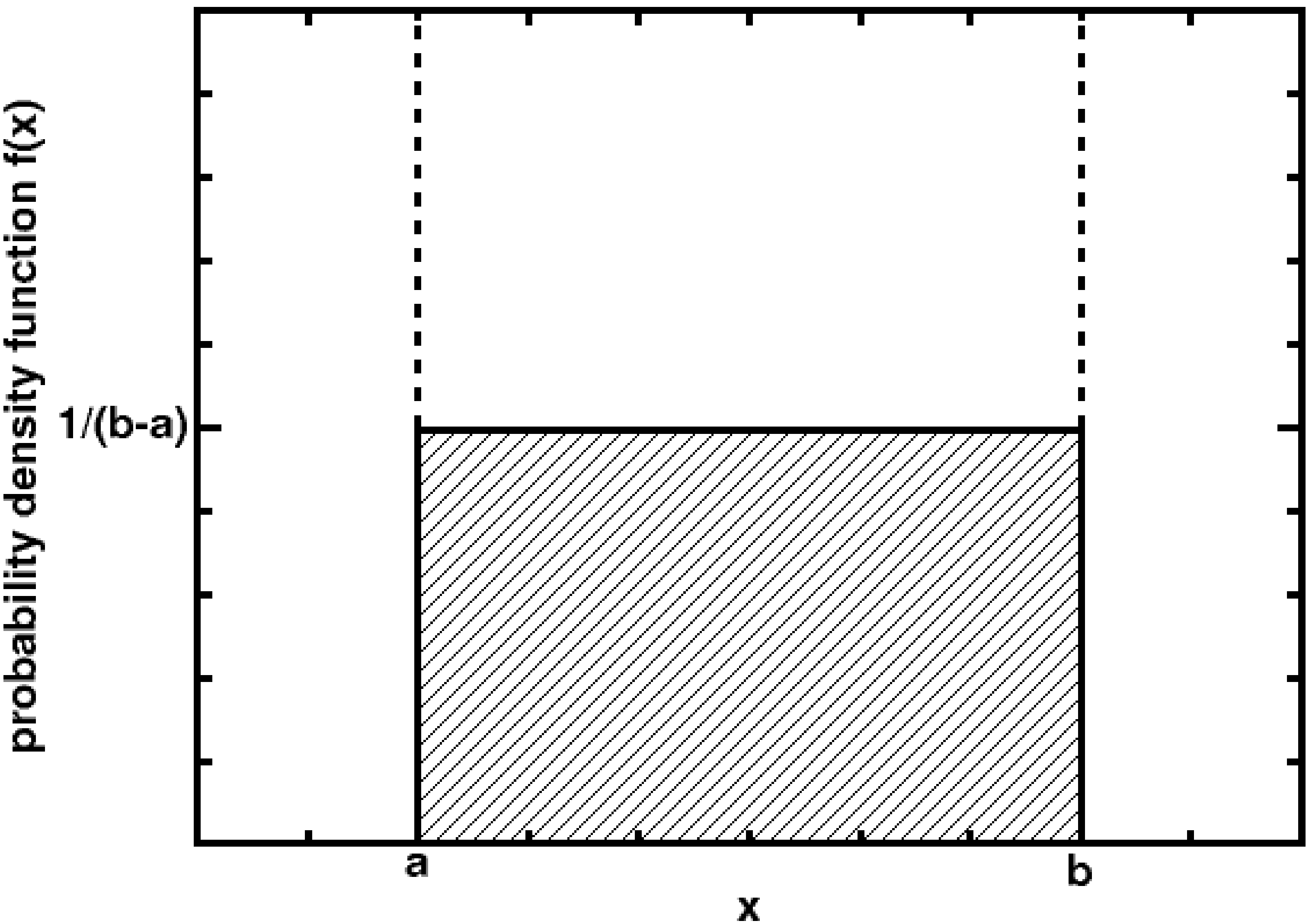
- For phase space source or full BEAM simulation. Makes electron range rejection really effective
- Can decrease simulation time by a factor of 5.5 for electron beams and of 3 for photon beams

Photon splitting

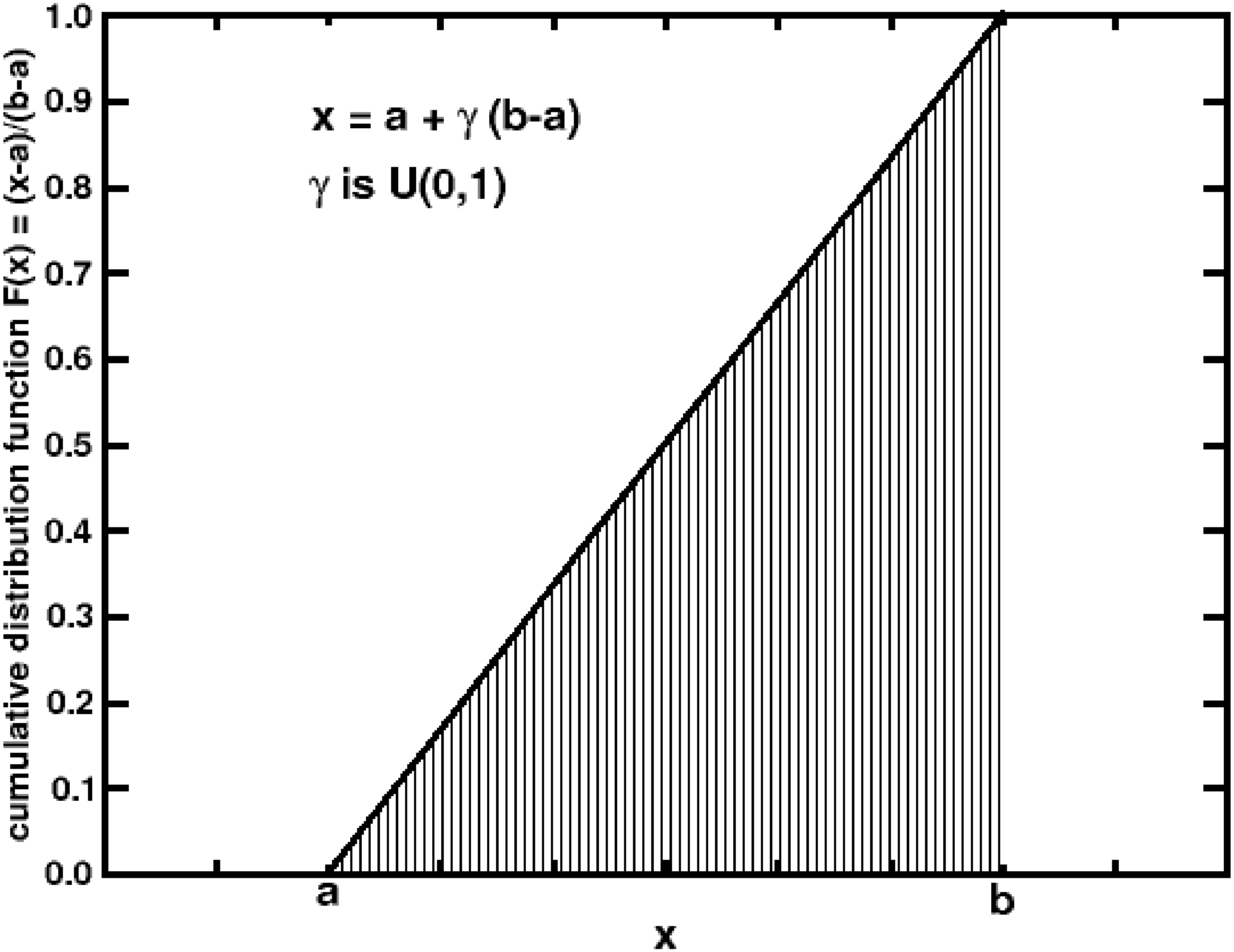
Main ideas:

- **Goal:** Increase efficiency of dose calculations for photon beams
- Split photon into N photons and sample their interaction point from N equidistant intervals of the exponential cumulative distribution function in the interval (0,1)
- Transport the N photons simulataneously across the geometry with one geometry tracing
- Play RR with scattered photons and split surviving fat photons
- First introduced for xVMC in Phys. Med. Biol **45** (2000) 2163.
- This technique is used in DOSXYZnrc, CAVRZnrc, and cavity.

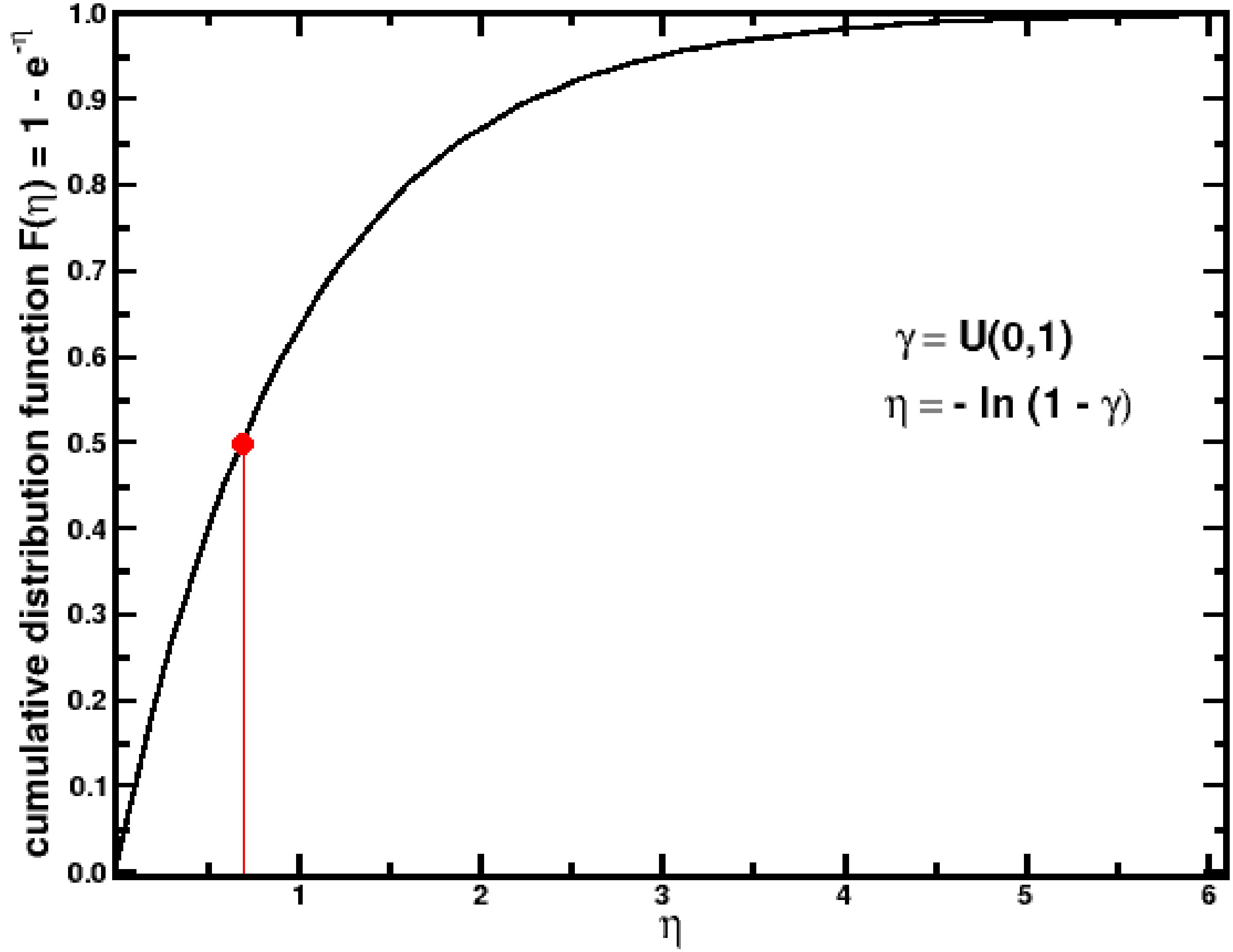
Uniform distribution



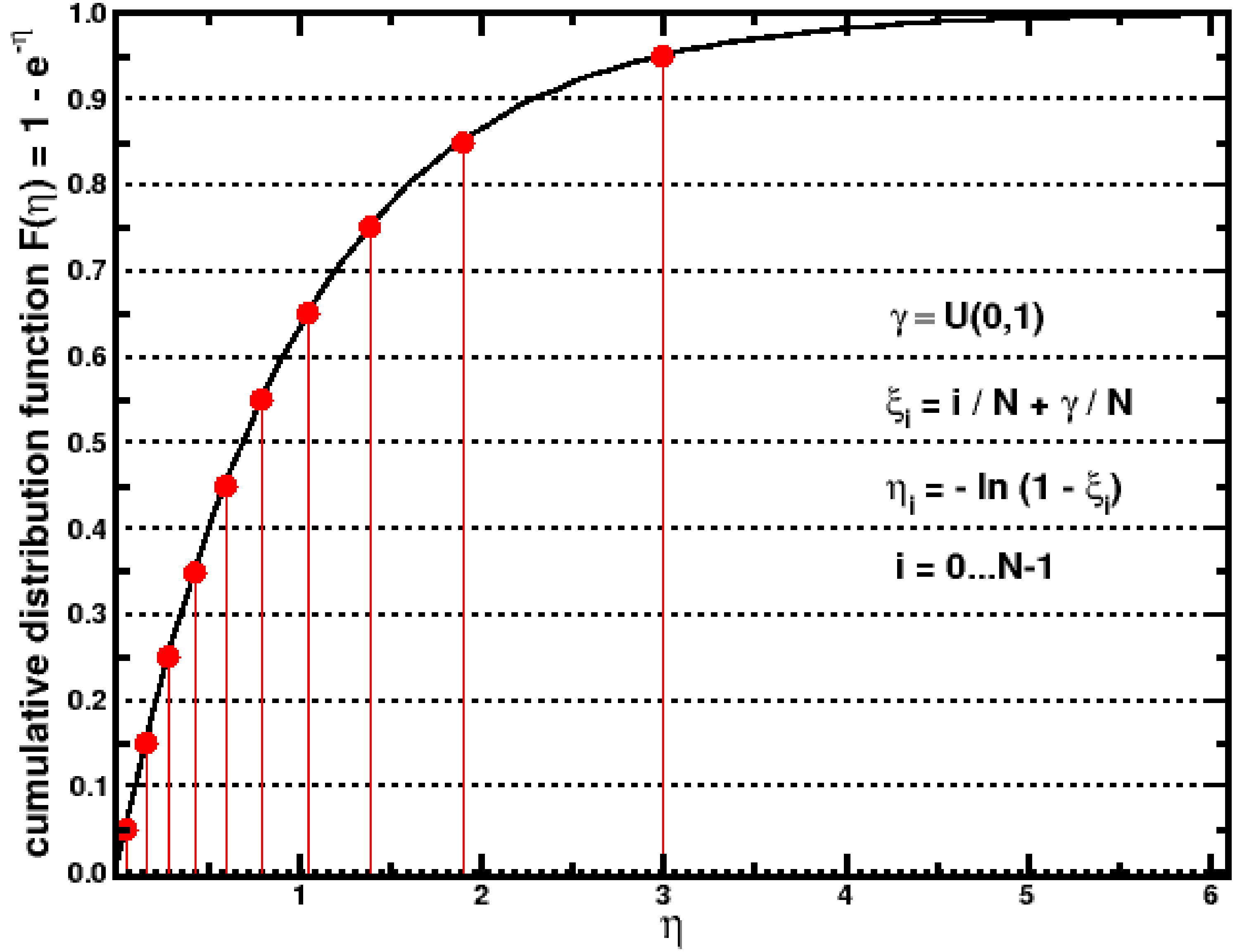
Sampling from the uniform distribution



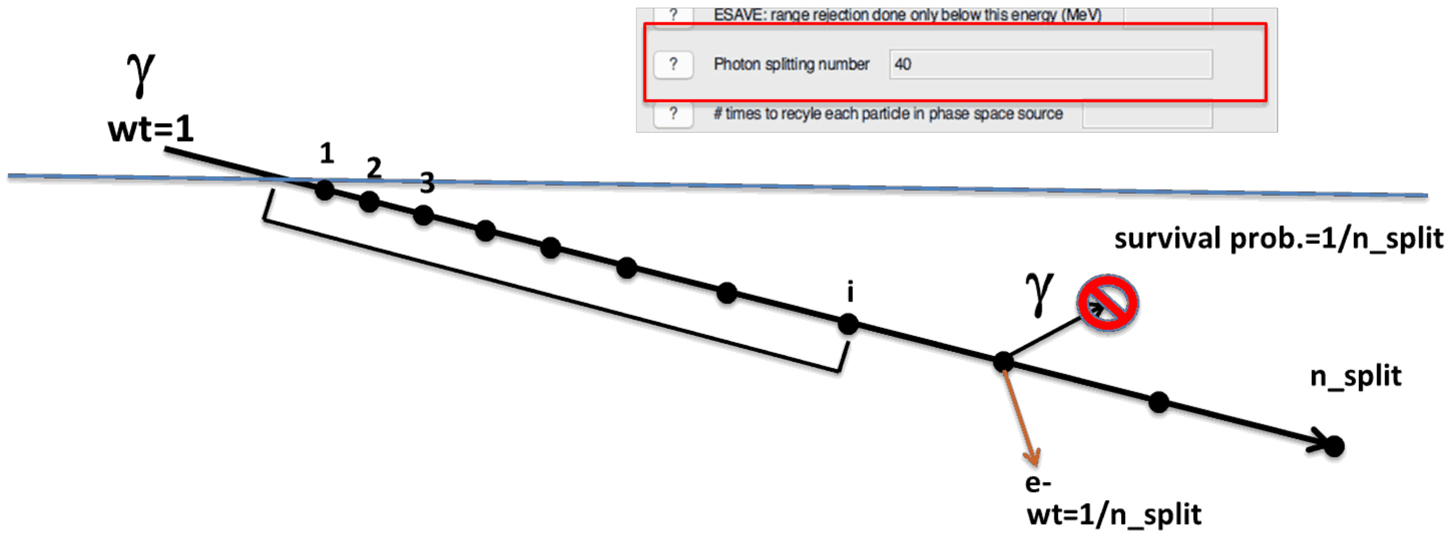
Sampling from the exponential distribution



Photon splitting



Photon splitting: Summary



- We have `n_split` interactions spread through the phantom and a single photon sets several electrons in motion.
- Electrons have weight $\frac{\mathtt{w}_0}{\mathtt{n\_split}}$, where $\mathtt{w}_0$ is the initial photon's statistical weight.
- RR scattered and bremsstrahlung photons $\Rightarrow$ survivors have weight $\mathtt{w}_0$ and are split again.

Choice of n_split

$$\texttt{n\_split} = \frac{n}{1 - e^{-X}}$$

where X is approximately equal to the number of γ-mfp in the geometry of interest and n $\geq$ 5. This will increase the number of primary interactions per incident photon by approximately `n_split`. See also [Kawrakow and Walters, Med. Phys. 33 (2006) 3046–3056]

The improvement in efficiency using splitting with photon beams is not great (25%) because photon transport is a relatively small fraction of the computing time, but can become a factor of 6 improvement if electron transport is turned off in the phantom.

Extremely useful if doing a DOSXYZnrc simulation with a BEAMnrc treatment head simulation source or if one includes the time spent for transport through the jaws in the efficiency balance.

Example: Photon splitting in DOSXYZnrc

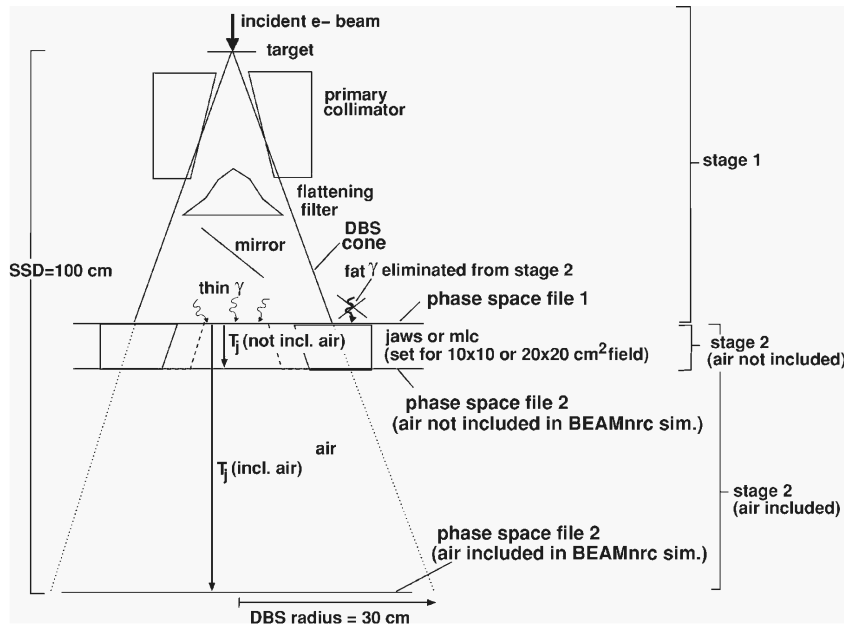
Approach A

- Create a phase space file above the jaws using BEAMnrc.

- Transport this phase space through the jaws for a given jaws setting and record a second phase space file just above the phantom.

- Use the phase space source in DOSXYZnrc to perform a dose calculation in a water phantom.

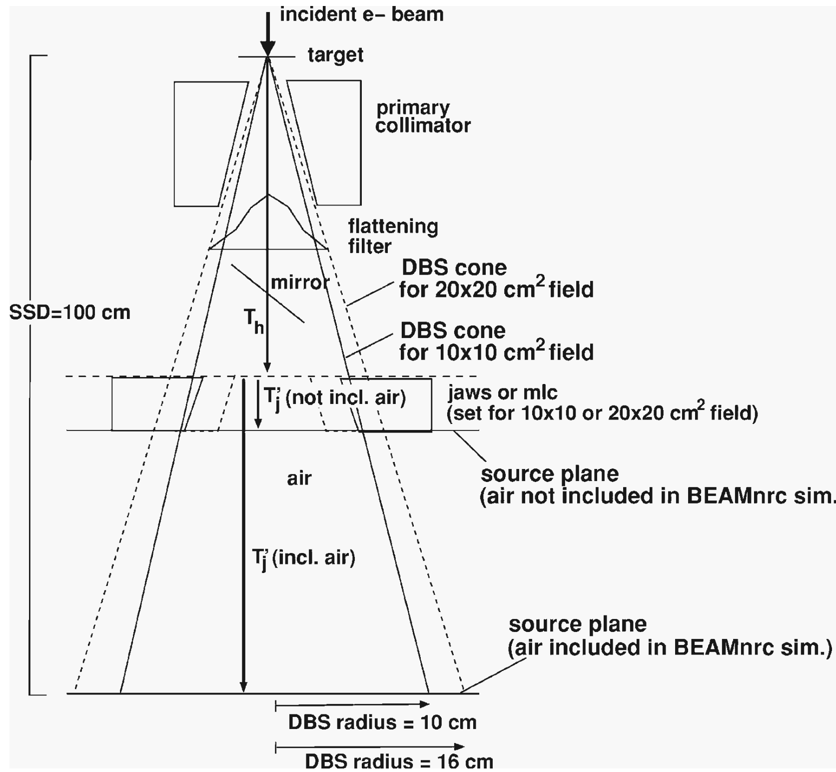
- Time to generate phase space above jaws can be excluded because it can be reused.



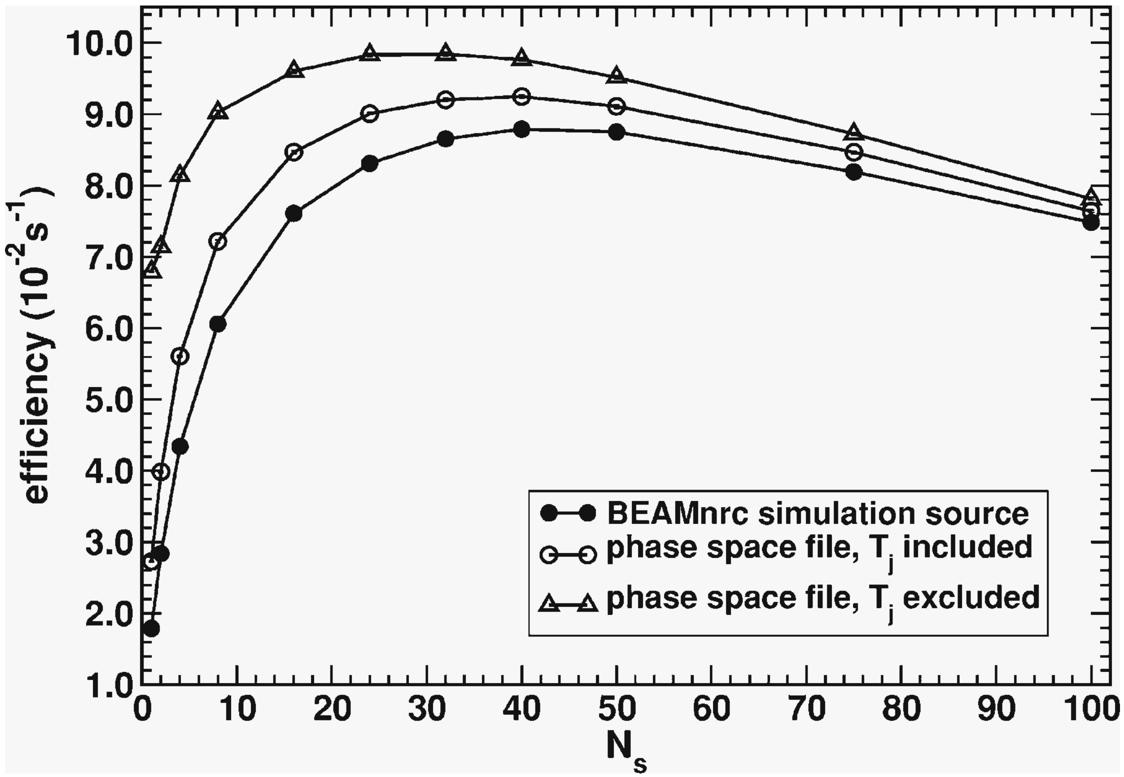
Example: Photon splitting in DOSXYZnrc

Approach B

- Run a DOSXYZnrc simulation using the BEAM simulation source, which performs a full treatment head simulation starting at the bremsstrahlung target.
- The CPU time in this case is the total time (DOSXYZnrc + treatment head simulation).



Example: Photon splitting in DOSXYZnrc

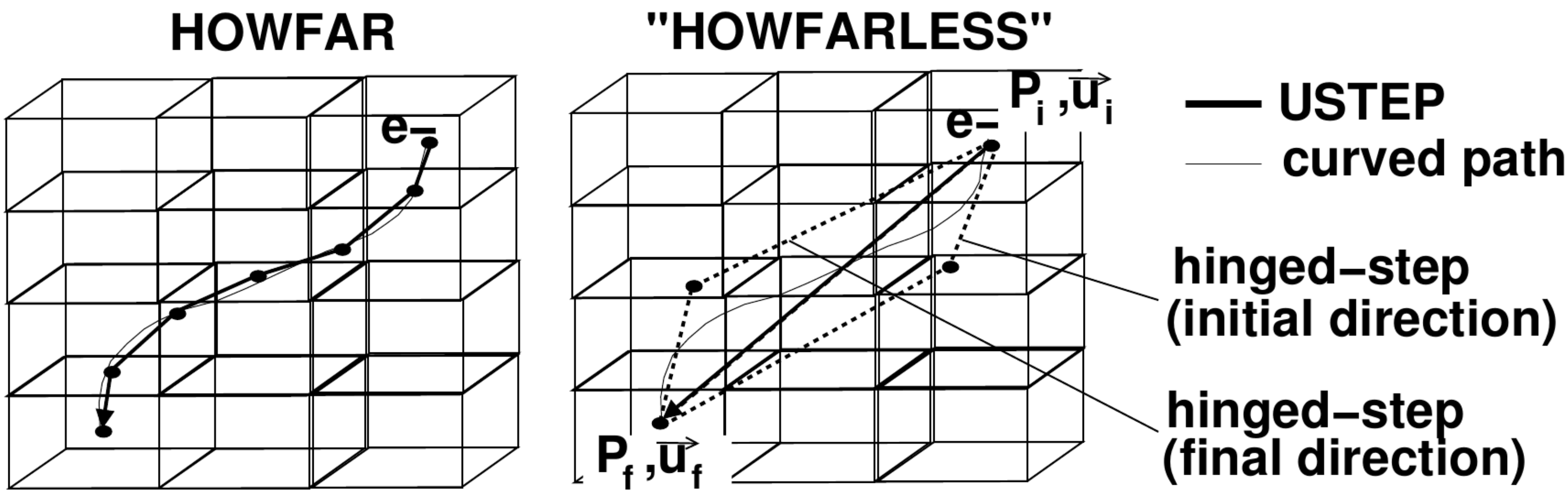


⇒ With a suitable choice of `n_split` for the DOSXYZnrc simulation and `NBRSPL` for the BEAM simulation the efficiency is comparable for A and B if initial time included in efficiency

⇒ Factor of $\sim$3 and $\sim$5 increase in efficiency by using splitting in DOSXYZnrc with phase-space and BEAMnrc sources, respectively. See [Med. Phys. 33 (2006) 3046–3056] for full details.

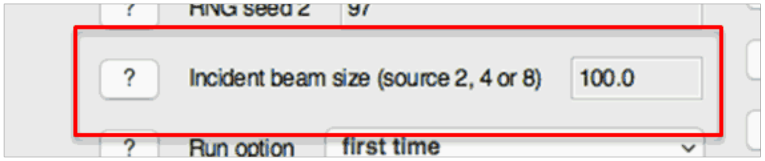
“HOWFARLESS” (`ihowfarless`=1)

Goal: Increase efficiency of *beam commissioning calculations* for homogeneous phantoms

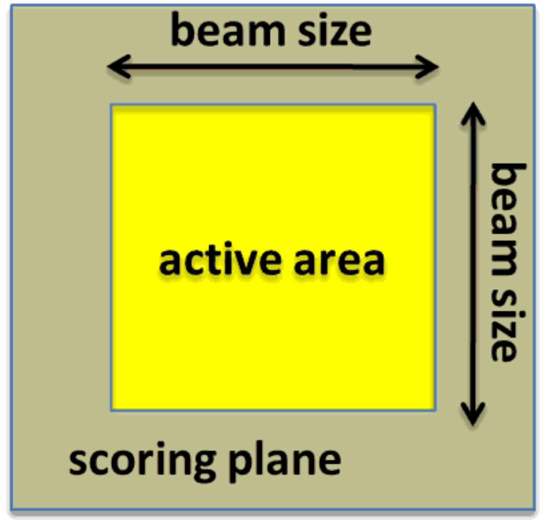


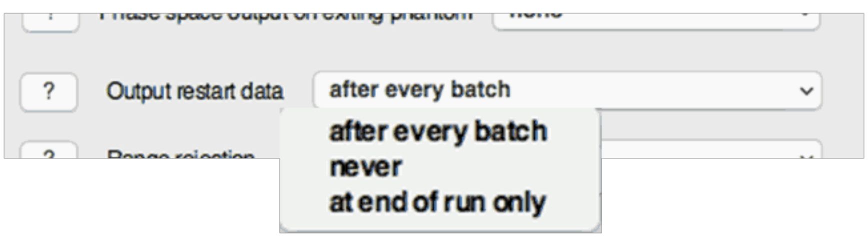
- HOWFAR and HOWNEAR only “consider” the outer boundaries of the phantom when determining step lengths ⇒ step length limited by `estepe`, max. energy loss/step
- 1:1 ratio of hinged steps based on initial:final direction
- improves efficiency by 50 %–90 % for BEAMnrc photon beams and by a factor of 3-5 for monoenergetic electron beams ⇒ highly-recommended

Incident beam size & Output restart data



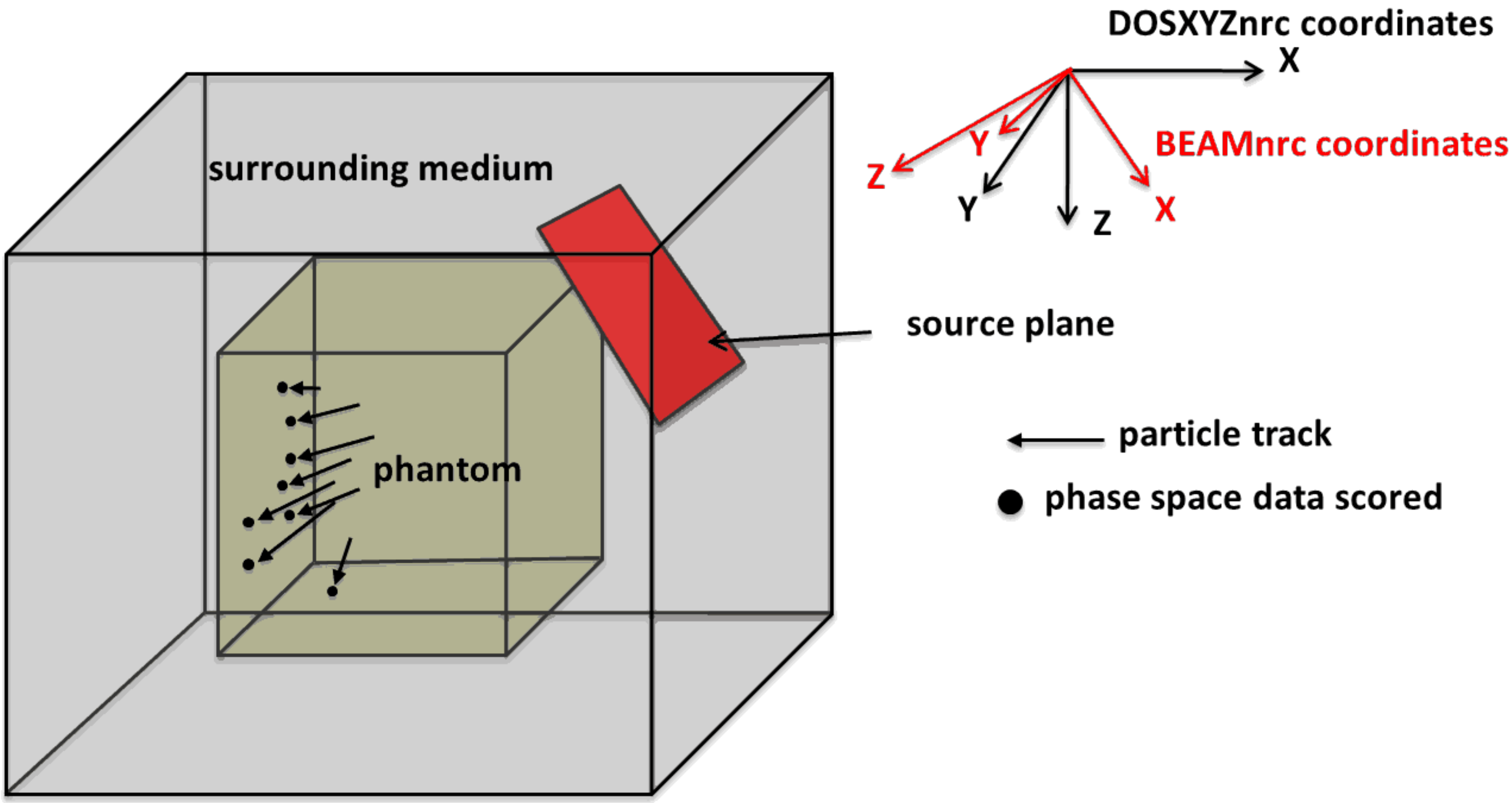
- only for phase space or BEAMnrc simulation source





- binary output to .egsdat file
- necessary for restarts (after crash or to improve statistics)
- but can become CPU time-limiting for phantoms with many voxels→output at end only

Phase space output (BEAMnrc or phase space source)



- can output in DOSXYZnrc or BEAMnrc coordinates
- 3-D phase space (X,Y,Z) scored for each particle →IAEA format only

DOSXYZnrc output

- *3ddose files: Information about the simulation geometry and the calculation results.
- *pardose files: Binary format output for parallel jobs containing enough information to reconstruct a *.3ddose file
- *egslst files: contain the dose (when asked for) and statistical data but also the information about simulation geometry, number of histories run, CPU time used, etc
- IAEA Phase space files (i_phsp_out)
  - i_phsp_out=1 ⇒ particle positions in DOSXYZnrc coordinate system
  - i_phsp_out=2 ⇒ particle positions in BEAMnrc coordinate system
- STATDOSE: Reads *.3ddose files
  - Visualizes dose profiles using xmgrace plots
  - Dose distributions can be normalized, rebinned
  - Statistical comparisons can be performed if only two or more dose distributions have been read in, and the voxel geometries are identical

DOSXYZnrc output (.3ddose file)

*.3ddose files are ASCII files which can be read by STATDOSE for analysis and plotting

block	data	description
1	nx ny nz	no of voxels in X,Y,Z
2	(xbound(i),i=1,nx+1)	X voxel boundaries
3	(ybound(j),j=1,ny+1)	Y voxel boundaries
4	(zbound(k),k=1,nz+1)	Z voxel boundaries
5	(((dose(i,j,k),i=1,nx),j=1,ny),k=1,nz)	Dose in Gy/incident particle or Gy/incident fluence
6	(((doseun(i,j,k),i=1,nx),j=1,ny),k=1,nz)	Fractional uncertainty on dose

Joint ICTP-IAEA Workshop on Monte Carlo Radiation Transport and Associated Data Needs for Medical Applications
28 October – 8 November 2024
ICTP, Trieste, Italy

Lecture 17

DOSXYZnrc dose calculations in a phantom

Blake Walters

Metrology Research Centre
National Research Council Canada

Government of Canada Gouvernement du Canada

Carleton University

Université de Montréal

nova scotia health