The Abdus Salam International Centre for Theoretical Physics
www.ictp.it

International Organization for Medical Physics

ICTP SChool On MEdical PHysics For RAdiation THerapy: DOsimetry And TReatment PLanning For BAsic And ADvanced APplications

13 - 24 April 2015

Miramare, Trieste, Italy

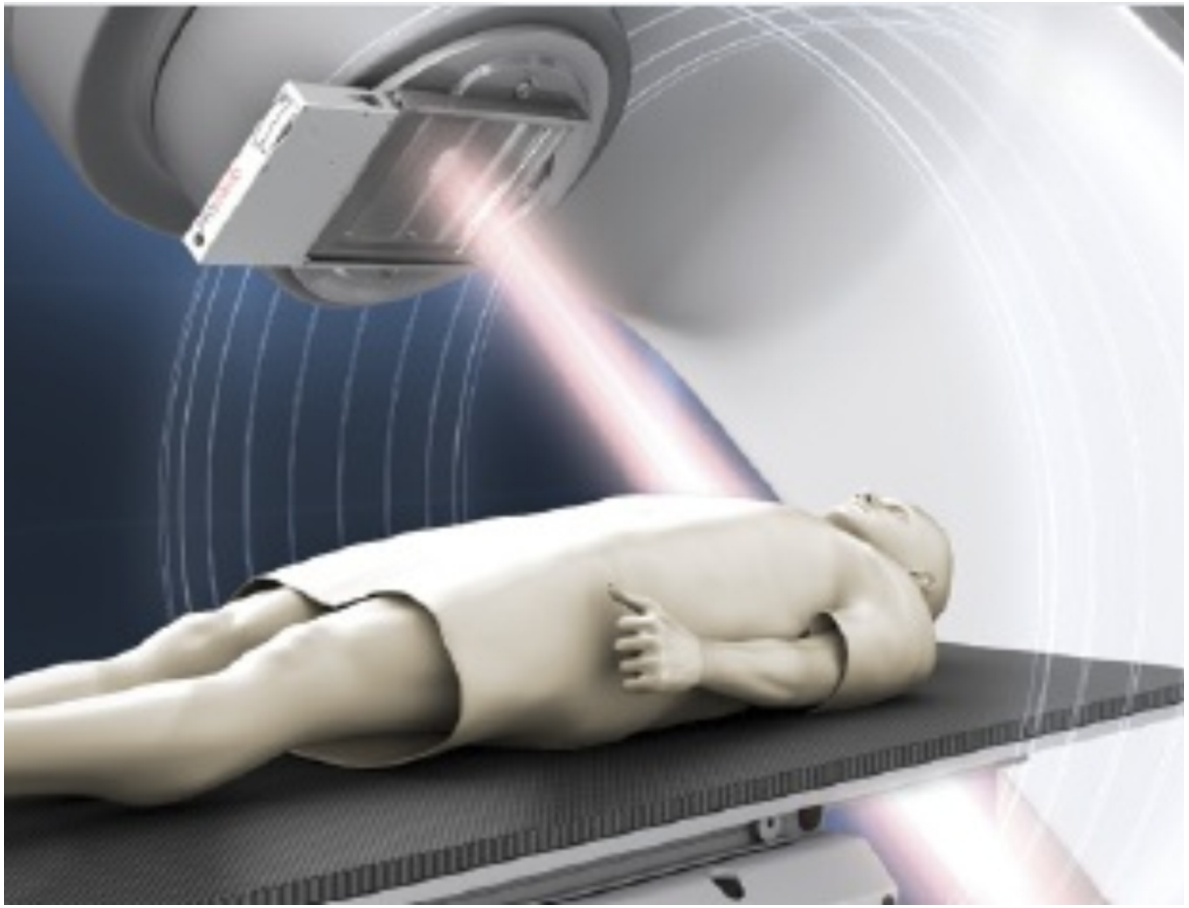
Treatment Planning Systems

G. Hartmann
EFOMP & German Cancer Research Center (DKFZ)
g.hartmann@dkfz.de

1. Introduction: Treatment Planning & dose calculation

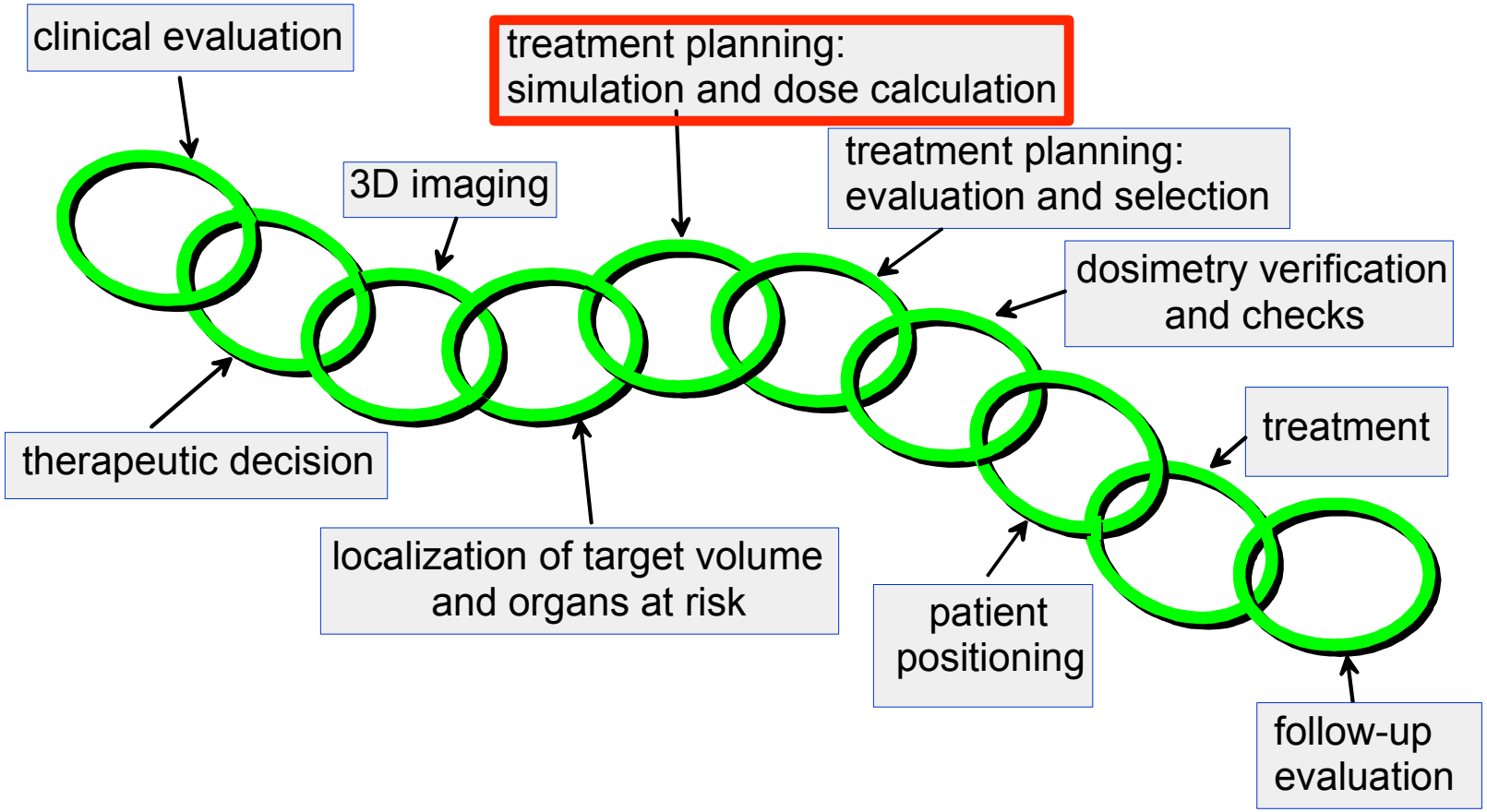
2. Key elements for a 3D dose calculation engine:
   - voxel model of the patient
   - beam model
   - ray tracing algorithm
   - dose calculation algorithm
   - optimization strategies
   - MC tracking

An idealistic picture showing a treatment with external radiation

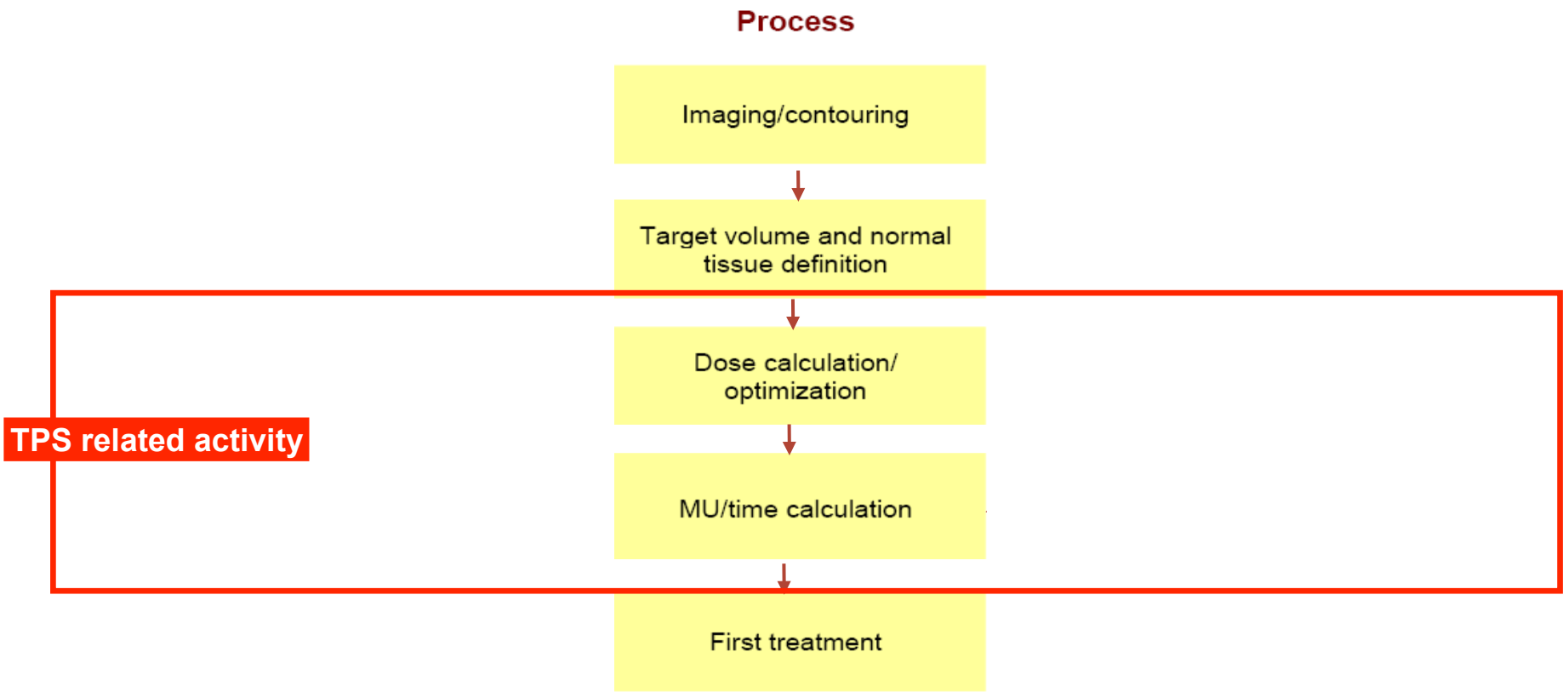


Delivery of a high dose of radiation requires thorough planning

Radiation delivery requires the whole process consisting of a chain of single procedures to be planned!



Steps of the treatment planning process, the professionals involved in each step and the QA activities associated with these steps (IAEA TRS 430)



This lesson deals explicitly with that component of the treatment planning process that makes use of the computer.

It is also frequently referred to as:

Computerized Treatment Planning.

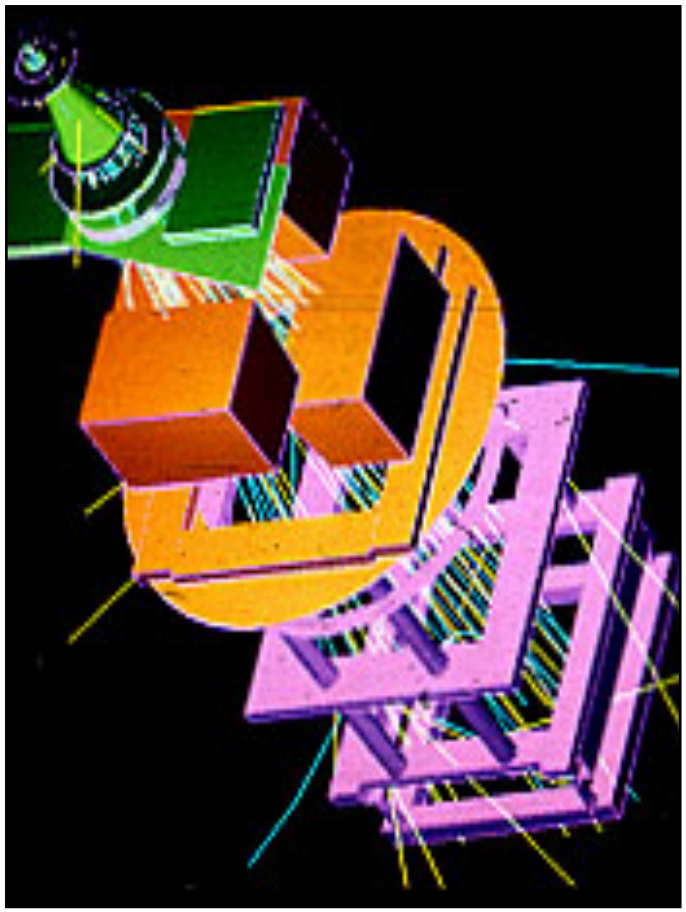
Such **Treatment Planning Systems** (TPS) are now always used in external beam radiation therapy and also in brachytherapy to generate beam shapes and dose distributions with the intent to maximize tumor control and minimize normal tissue complications.

Main elements of a TPS

1. Import of patient data (DICOM Format)
2. Establishment of the beam model
3. Generation of the individual patient model
4. Definition of target volume(s) and OARs
5. Definition of irradiation parameters
6. Dose calculation
7. Plan evaluation, Optimization
8. Dose prescription and determination of monitor units
9. Export of treatment parameters
10. Documentation

Imaging part

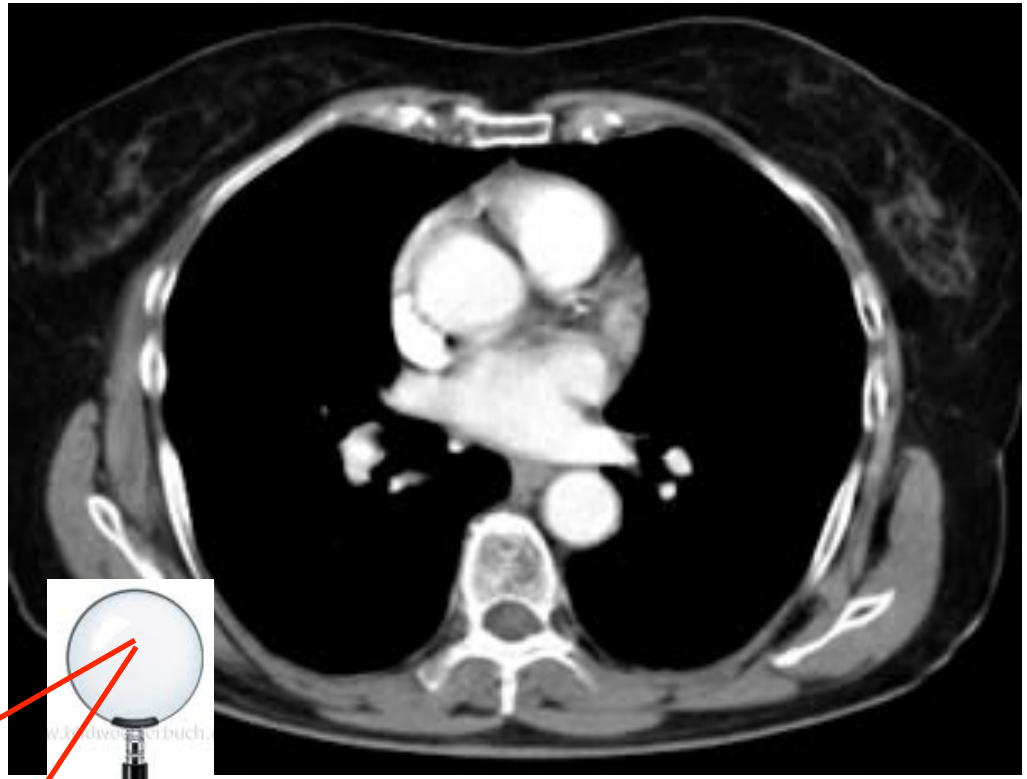
Dose calculations have evolved from simple 2D models through 3D models to 3D Monte-Carlo techniques, and increased computing power continues to increase the calculation speed.

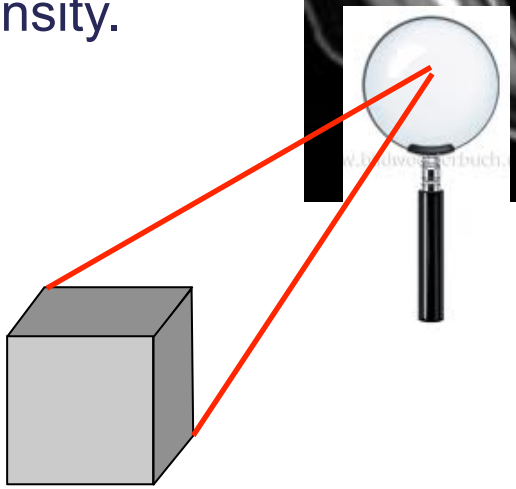


Monte Carlo simulation of an electron beam produced in the accelerator head.

Voxel model of the patient

From a series of CT images we can establish a patient model that consists of cuboidal blocks each with an individual density.





These cuboidal blocks are normally referred to as voxels

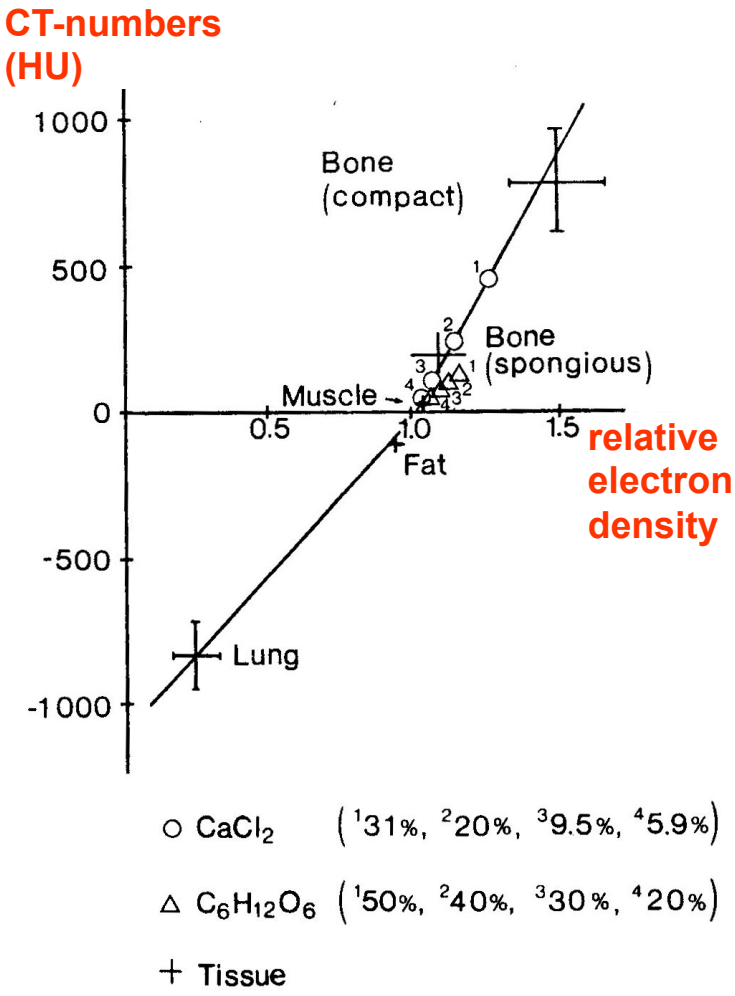
Voxel model of the patient

In order to adjust the dose calculation to an individual patient, we need:

the contours of patient, CTV, and anatomical structures

the information of tissue inhomogeneities.

Inside the patient, the **relative electron density** of each voxel can be determined from the patient CT data set.

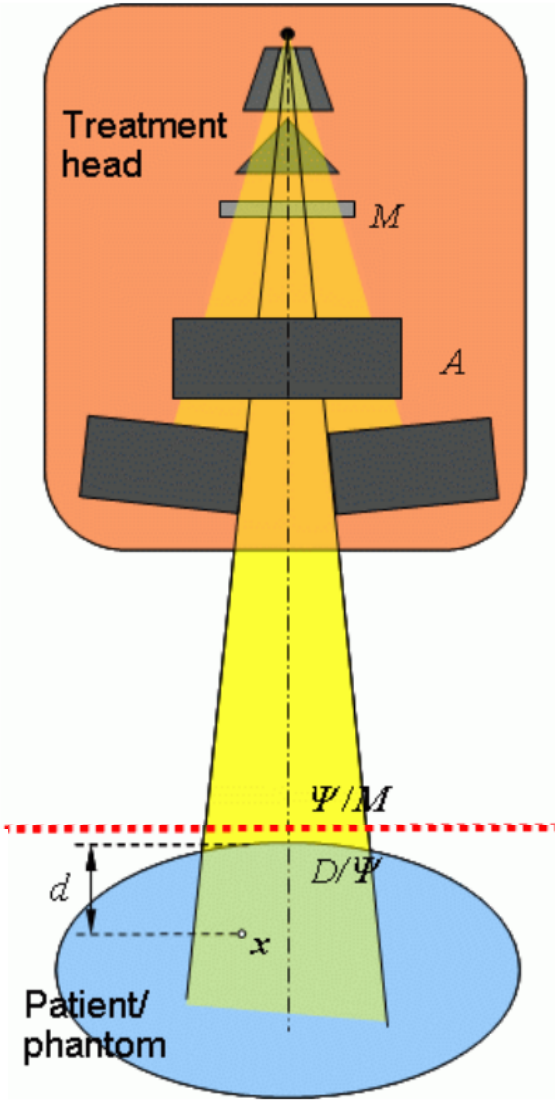


Beam model

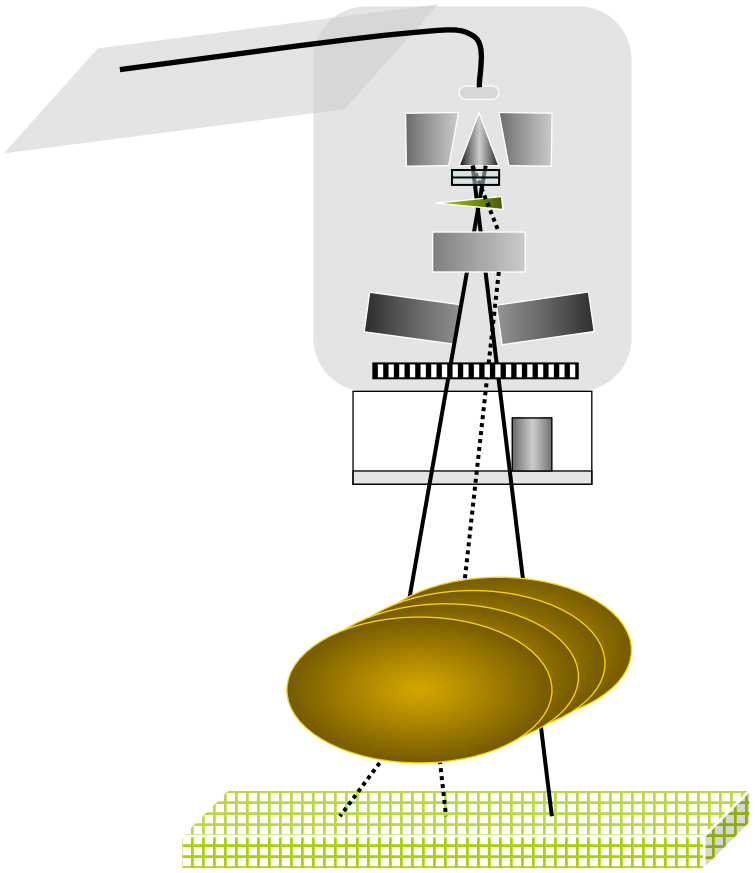
The modern approach utilizes the natural divider between

- the radiation sources inside the treatment head

- and the patient or the phantom.



Beam model: treatment head



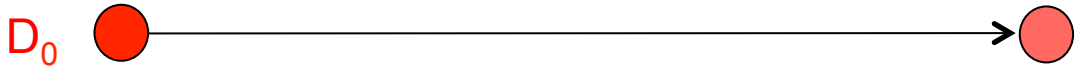
Schematic drawing of an accelerator head (from A. Ahnesjö)

A complete model requires:

- Finite photon source size
- Open fluence distribution
- Fluence modulation
  - Step&shot
  - Dynamic
  - Wedges
- Head scatter sources
  - flattening filter
  - collimators
  - wedges
- Monitor back scatter
- Collimator leakage, including
  - MLC interleaf leakage
  - shape of MLC leaf ends
- Beam spectra
- Spectral changes
- Electron contamination

Beam model and ray tracing

A (rather simple) method of dose calculation:



The dose D_0 is known
at a certain point P_0 at the surface

For a ray of photons:

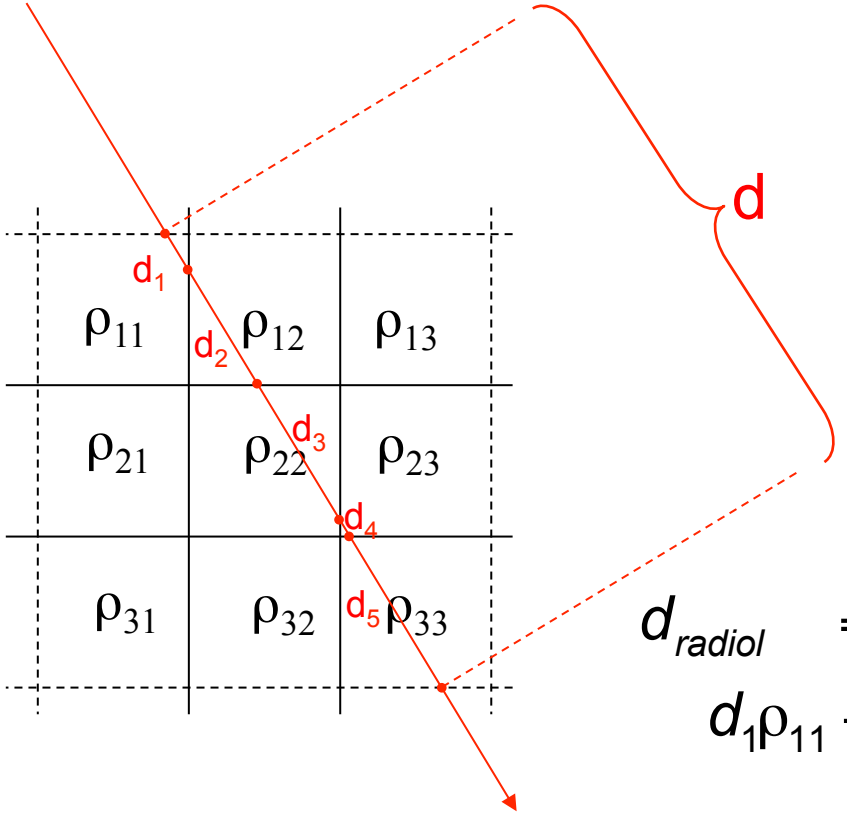
$$D_1 = D_0 \cdot e^{-\mu d}$$

Where d is the radiological path from P_0 to P_1

If this method is applied within a voxel array, it is frequently referred to as **ray tracing**

Ray tracing

The term “Ray tracing” is frequently used to determine the **radiological path length** through a voxel array representing a patient (with relative densities ρ_{11}, ρ_{12}, ρ_{13}, …).



The geometrical path d within the patient:

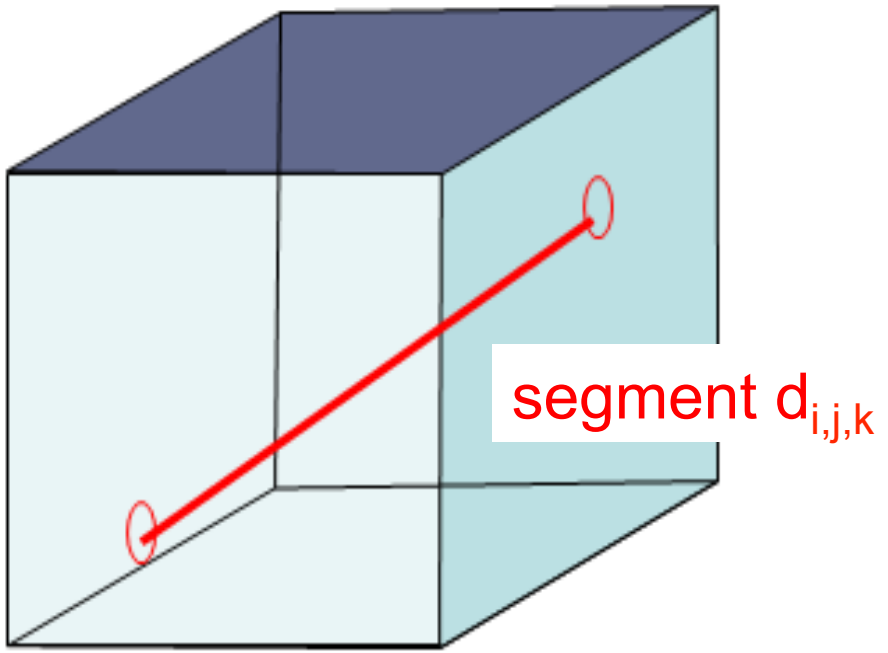
The radiological path d_{radiol} within the patient (simplified):

$$d_{radiol} = d_1\rho_{11} + d_2\rho_{12} + d_3\rho_{22} + d_4\rho_{23} + d_5\rho_{33}$$

Ray Tracing

In order to determine the radiological path d_{radiol} through the patient, one has to determine – voxel by voxel – the segments d_{ijk} in each single voxel I,j,k in the 3D space.

Consider a
voxel with index i,j,k



Ray Tracing

In a general formulation, the radiological path d_{radiol} is

$$d_{radiol} = \sum_i \sum_j \sum_k d_{i,j,k} \cdot \frac{(\text{interaction coefficient})_{i,j,k}}{\text{interaction coefficient}_{water}}$$

For photons:

$$d_{radiol} = \sum_i \sum_j \sum_k d_{i,j,k} \cdot \frac{\mu_{i,j,k}}{\mu_{water}}$$

It is obvious that the evaluation of this equation scales with the number of voxels = $N_i \times N_j \times N_k$

(for instance: 256 x 256 x 64 = 4 10^6 iterations

Ray Tracing

However, there are algorithms of ray tracing which are **much** faster:

Fast calculation of the exact radiological path for a three-dimensional CT

Robert L. Siddon

Med. Phys. 12 (2), Mar/Apr 1985

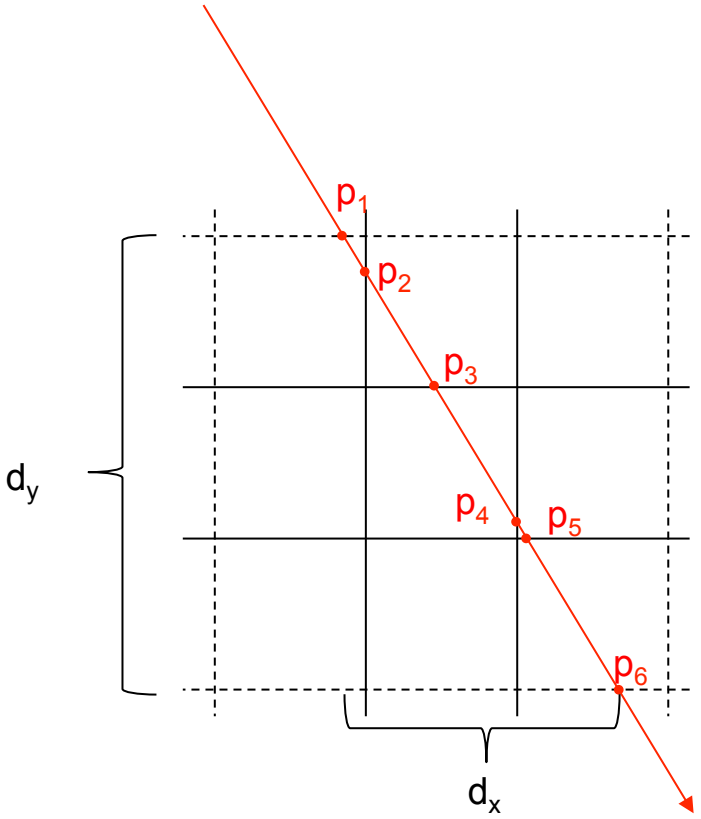
Fast Algorithm for computer control of a digital plotter CT

J. E. Bresenham

IBM Systems Journal Vol.4 No. 1 1965

Ray Tracing: Siddon's algorithm (illustrated in 2D)

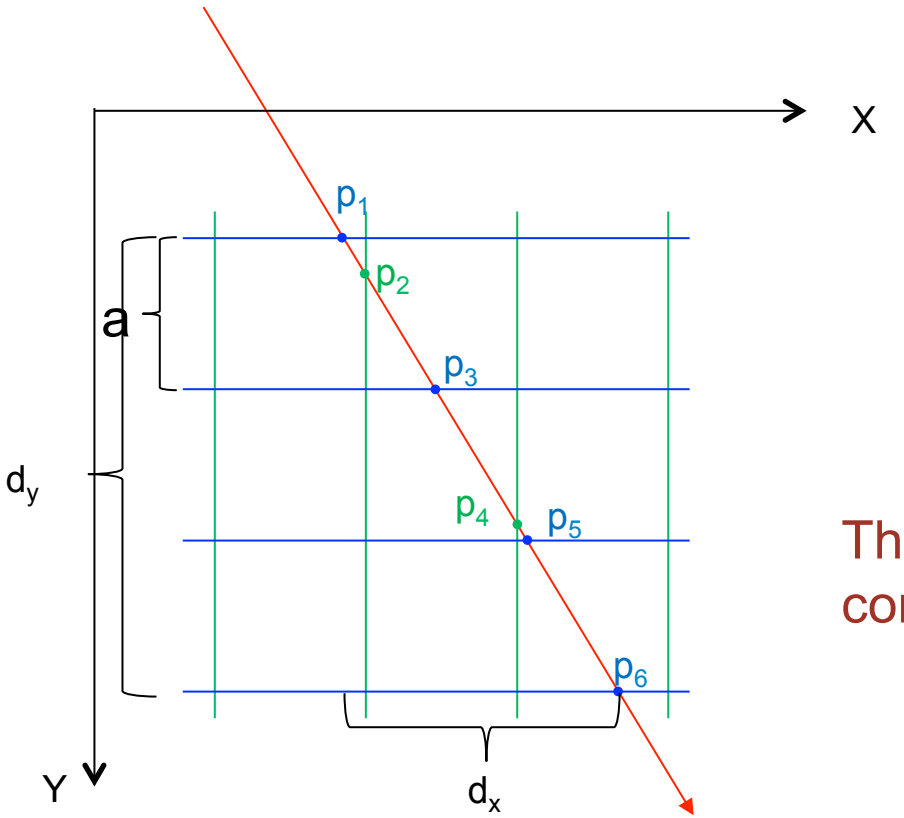
Consider the intersection points of the geometrical path d:



$$d_{geometrical} = \sqrt{(d_x)^2 + (d_y)^2}$$

Ray Tracing: Siddon's algorithm (illustrated in 2D)

…… as being intersections with the equally spaced vertical and horizontal lines (distance: a) in blue and green:



X coordinates of the intersection points:

$$x_{i=2,4} = x_1 + \alpha_{x,i} \cdot d_x$$

$$\alpha_{x,i} = (x_i - x_1)/d_x$$

Y coordinates of the intersection points:

$$y_{i=1,3,5,6} = y_1 + \alpha_{y,i} \cdot d_y$$

$$\alpha_{y,i} = (y_i - y_1)/d_y$$

The $\alpha_{x,i}$ and $\alpha_{y,i}$ can be merged into a common series of increasing values:

$$\{\alpha\} = \{\text{merge}[\alpha_{x,i}, \alpha_{y,i}]\}$$

$$\{\alpha_1, \ldots., \alpha_m, \ldots, \alpha_6\}$$

Ray Tracing: Siddon's algorithm

Therefore the individual distance d_m can be calculated as:

$$d_m = d \cdot \left[\alpha_m - \alpha_{m-1}\right] \qquad \text{with} \qquad d = \sqrt{(d_x)^2 + (d_y)^2}$$

In a similar way, the **indices** of each voxel i and j can be also obtained from the sequence of $\{\alpha_1, \ldots., \alpha_m, \ldots, \alpha_6\}$

$$(i\ m) = \text{integer}\left(1 + \left[x_{16} \cdot \frac{\alpha_m - \alpha_{m-1}}{2}\right] \Big/ a\right)$$

$$(j\ m) = \text{integer}\left(1 + \left[y_{16} \cdot \frac{\alpha_m - \alpha_{m-1}}{2}\right] \Big/ a\right)$$

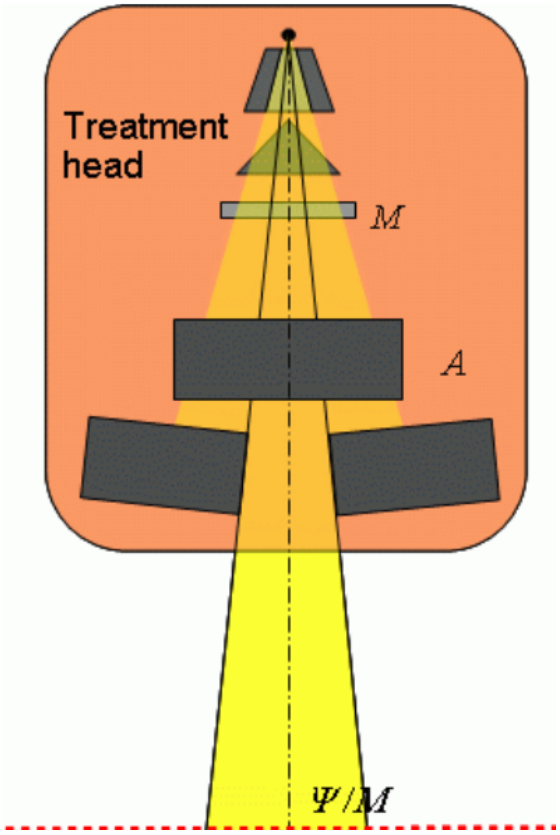
Ray Tracing: Siddon's algorithm

The charm of this algorithm is:

It does not scale with the number of **voxels** N_i x N_j x N_k but with number of **planes** $(N_i+1)+(N_j+1)+(N_k+1)$.

For instance:
Instead of 256 x 256 x 64 = 4 million iterations we need only (256+1)+(256+1)+(64+1) = 579 iterations

Beam model: treatment head



Terma $\int \Phi_E \cdot \left(\frac{E\mu}{\rho} \right) \cdot dE \quad \left[\frac{J}{kg} \right]$

Kerma $\int \Phi_E \cdot \left(\frac{E\mu_{tr}}{\rho} \right) \cdot dE \quad \left[\frac{J}{kg} \right]$

Collision Kerma $\int \Phi_E \cdot \left(\frac{E\mu_{en}}{\rho} \right) \cdot dE \quad \left[\frac{J}{kg} \right]$

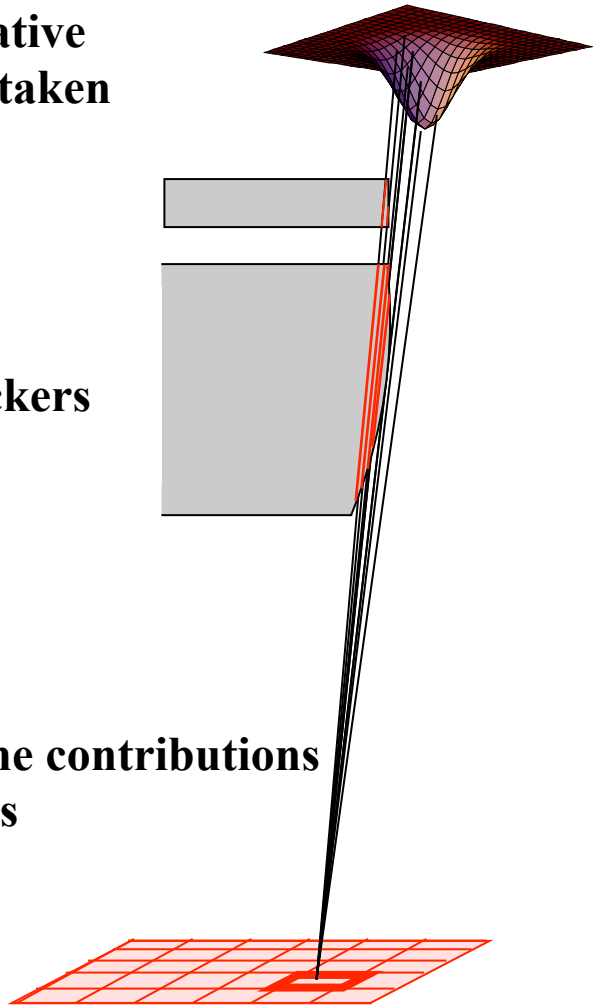
A “Fluence engine“ would provide the required knowledge to calculate, for instance collision kerma

The beam model can also be considered as a fluence engine:

The width, shape and other radiative properties of the source must be taken into account

Collimators can be raytraced, or approximated as ideal beam blockers

For each element, find the contributions from the relevant sources



Calculate the value of a fluence matrix element

Dose calculation algorithm

Dosimetrical quantity	Principle of calculation	Required methods & ingredients
„dose“	ray tracing	ray tracing algorithm
Collision Kerma	Fluence times energy absorption coefficient	determination of fluence differential in energy
Absorbed dose	solution of Boltzmann Transport Equation, superposition	Monte Carlo code, superposition algorithm

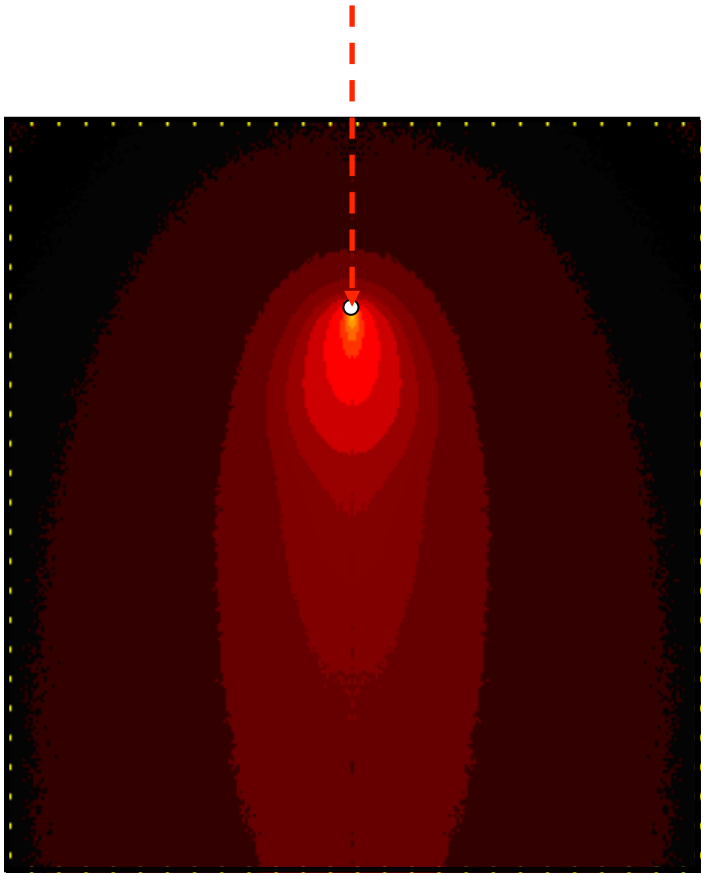
Superposition and Point kernel

What is a point kernel?

Imagine a water absorber and a **point** at a certain depth.

Imagine that many photons are coming all along a vertical path and are all **interacting at this point only**.

A point kernel represents the energy transport and dose deposition of secondary particles stemming from **that point of interactions**.

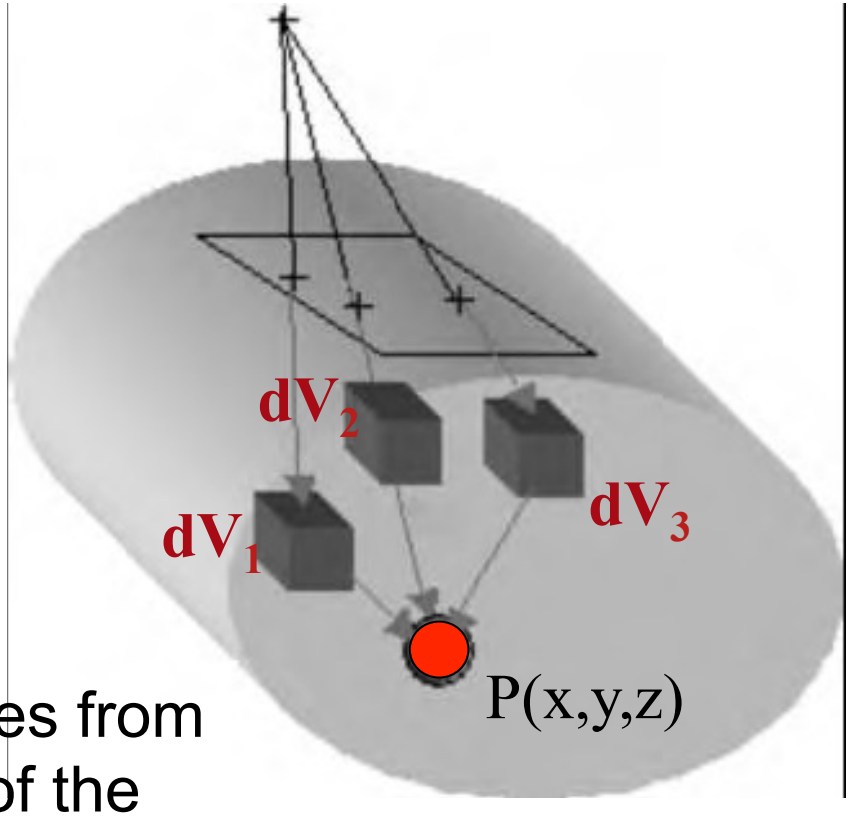


Point kernels are extremely useful for the superposition method.

The superposition principle is summarized in the following Figure:

The dose at a point P(x,y,z) can be considered as the sum of the contributions of the energy launched at a distance from P

i.e. in volume elements $dV(x_0,y_0,z_0)$.



This elementary energy originates from the **energy fluence $p(x_0,y_0,z_0)$** of the primary photons impinging on dV and the photon interactions within dV.

We denote the scatter energy per unit primary photon fluence launched at dV and reaching P as:

$$s(x,x_0,\ y,y_0,\ z,z_0)$$

Then the dose at P(x,y,z) is

$$D_P(x,y,z) = \iiint_V p(x',y',z')\ s(x,x',y,y',z,z')\ \mathrm{d}V$$

fluence at x',y',z'

scattered energy from x',y',z' absorbed at x,y,z

We can summarize this by the following statement:
The dose deposition is viewed as a superposition of appropriately weighted responses to point irradiations.

These responses are referred to as **point kernels**.

These kernels usually are not accessible through measurements but can be calculate by use of Monte Carlo particle transport codes (example).

Under conditions where the kernels are spatially invariant, the superpositions can be efficiently evaluated by means of convolutions.

Dose calculation methods

There are various methods of kernel implementation:

point kernel

pencil kernel

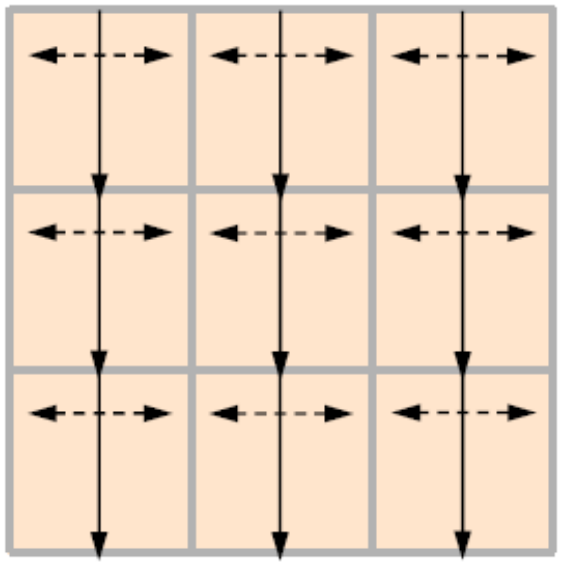
collapsed cone

etc. ….

Dose deposition approximations

Kernel/range scaling due to non-water like media.

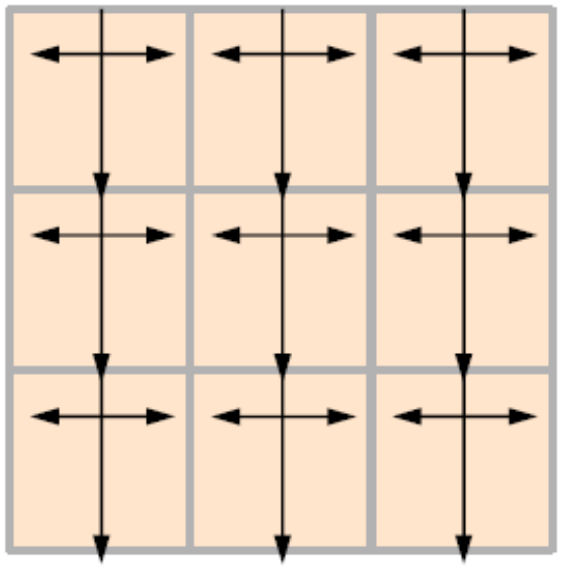
Standard Pencil beam



Energy transport: Along rayline and laterally.

Kernel scaling by radiological pathlengths: Only along rayline.

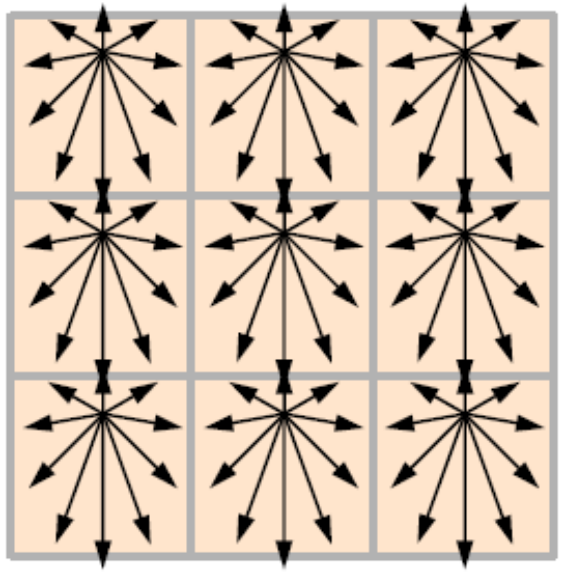
Varian AAA
(Anisotropic Analytical Algorithm)



Energy transport: Along rayline and laterally (16 directions).

Kernel scaling by radiological pathlengths: In all (17) directions.

Convolution superposition/ Collapsed cone



Energy transport: Along approx. 100 directions.

Kernel scaling by radiological pathlengths: In all directions.

Dose calculation methods

There are various methods of kernel implementation:

point kernel

pencil kernel

collapsed cone

etc. ….

Each of them has advantages and disadvantages, in particular when applied to structures with lateral borders and such with low density (Lung).

Optimization

Examples:

Which treatment parameters can/should be optimized:

In IMRT: Intensity maps for each beam

or

weights of beams segments

Further parameters:

Beam angles

Number of beams

Type of radiation

Energy

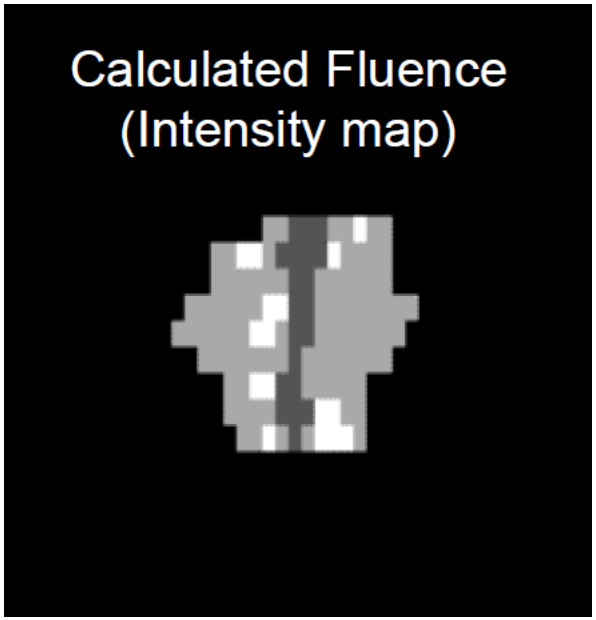
Optimization

What is needed in IMRT:

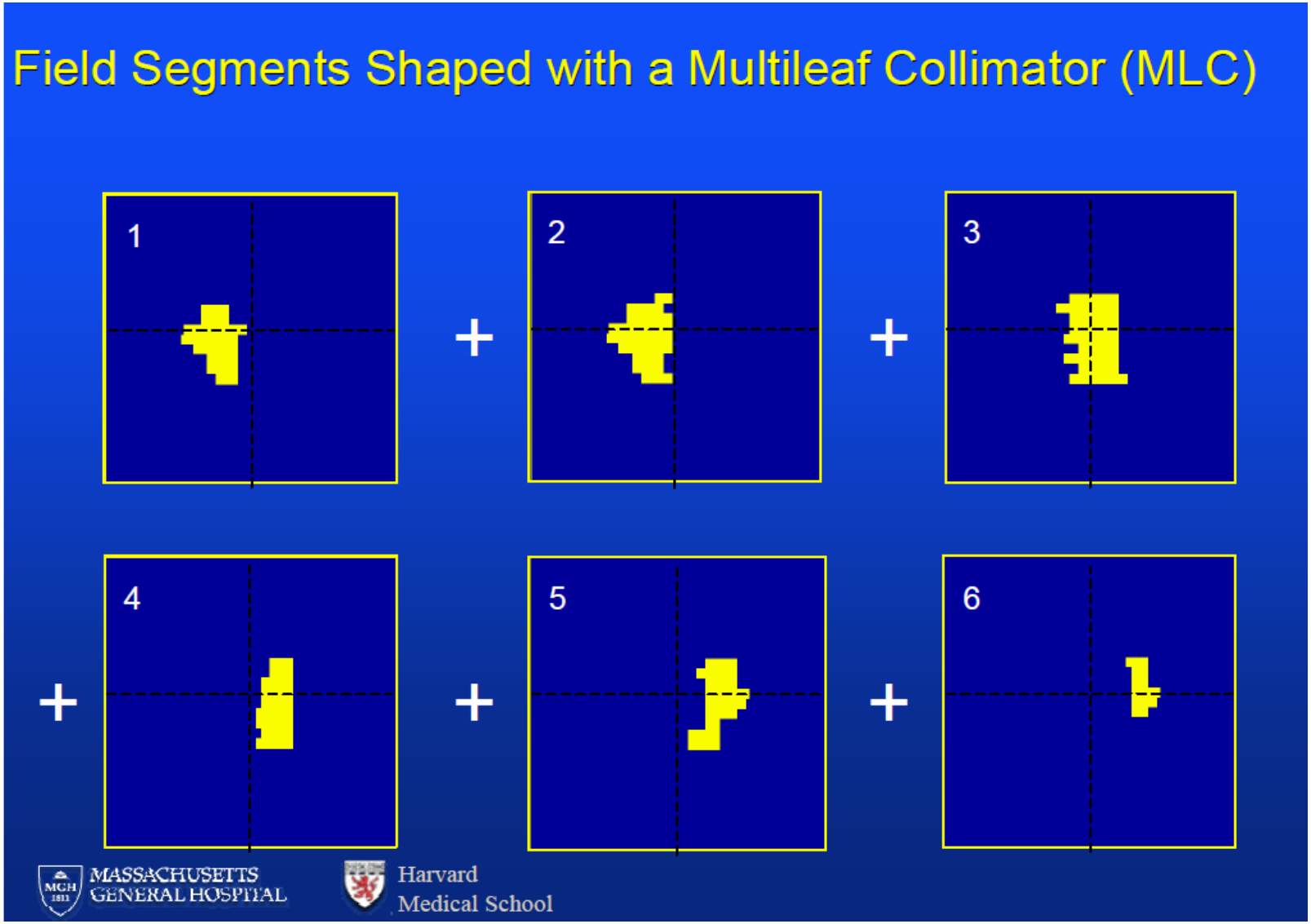
Intensity maps for each beam

or

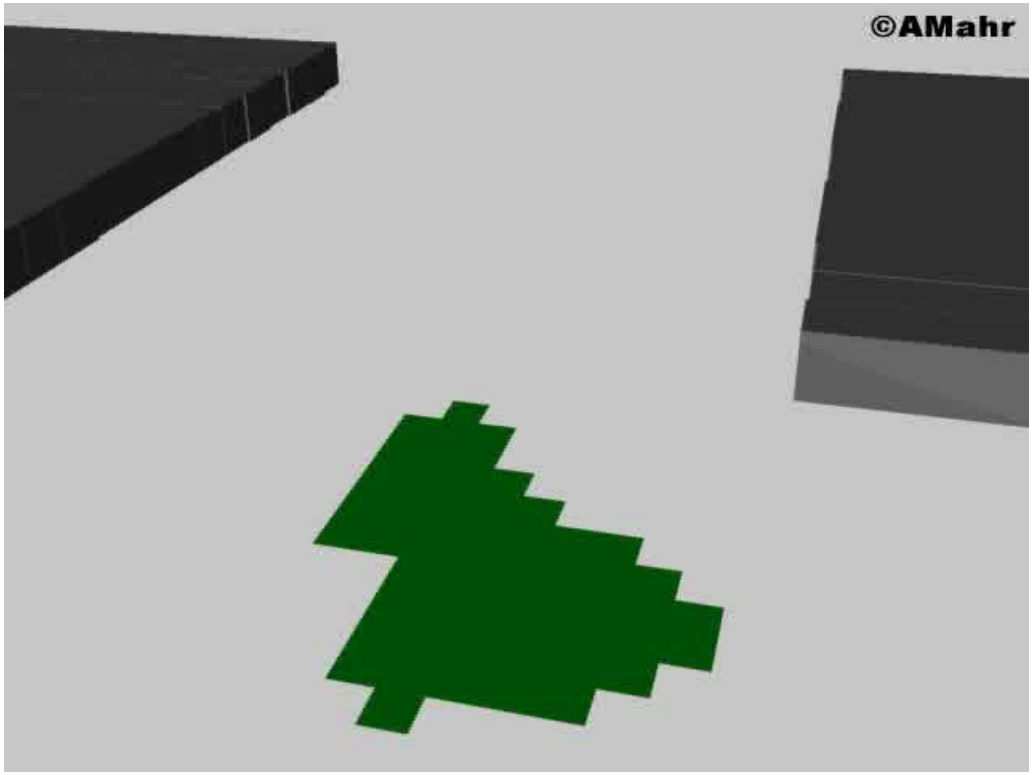
weights of beams segments



Key elements for a 3D dose calculation engine:
Optimization/IMRT



Key elements for a 3D dose calculation engine:
Optimization/IMRT



Optimization

Objective Functions - Example

$$F_k(\vec{x}) = w_k \sum_{j=1}^{N_k} \left(C_+ \{ D_{k,j}(\vec{x}) - D_k^{\max} \} \right)^2$$

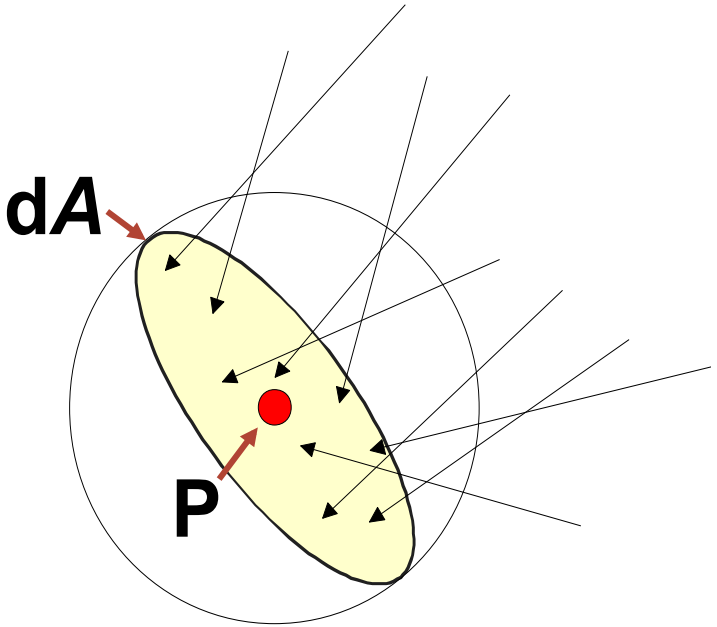
- w_k: weight
- $D_{k,j}(\vec{x})$: dose at voxel j in OAR k
- $D_k^{\max}$: tolerance dose

Constraint operator, e.g.: $C_+\{a\} = \begin{cases} a & \text{for} \quad a \geq 0 \\ 0 & \text{otherwise} \end{cases}$

MASSACHUSETTS GENERAL HOSPITAL

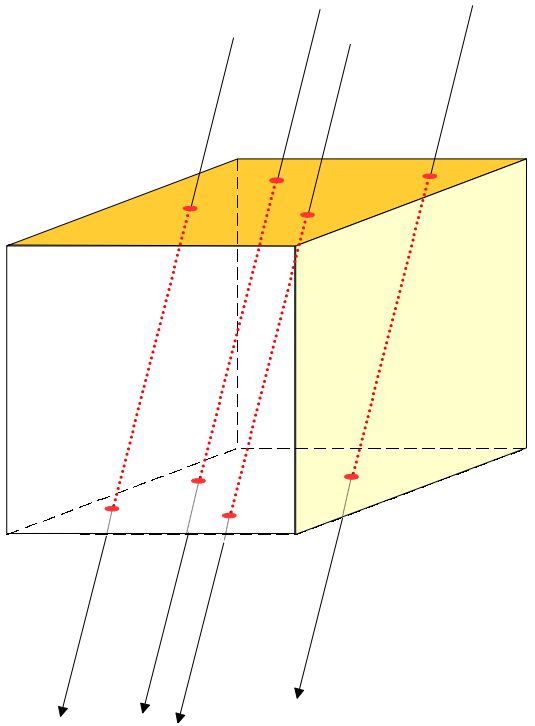
Harvard Medical School

Fluence and tracking



$$\Phi(r) = \frac{dN}{dA}$$

Alternative definition:



$$\Phi(\vec{r}) = \frac{dL(\vec{r})}{dV}$$

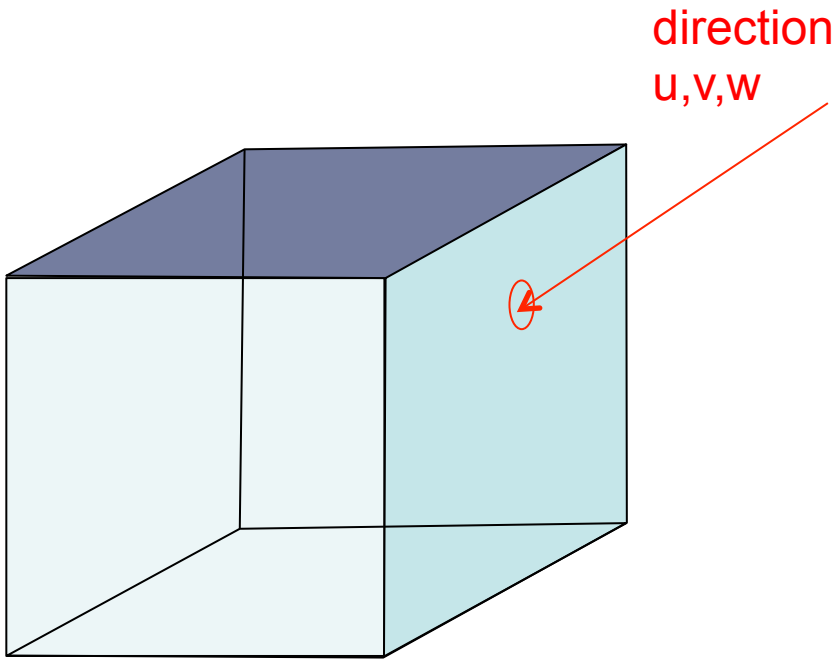
Monte Carlo simulations of particle transport processes are a faithful simulation of physical reality because:

- particles are “born” according to distributions describing the **source**,
- they **travel certain distances:**
  a) to the next point of interaction, or
- b) going through the entire voxel without an interaction
- **scatter into another energy and/or direction** according to the corresponding differential cross section, **possibly producing new particles** that have to be transported as well.

This methods requires a tracking of each individual particle through a certain geometry, and the summation over a large number of particles.

Individual particle tracking within the Monte Carlo method

The **path length** within a volume of interest and thus the fluence can be determined by the following procedure:



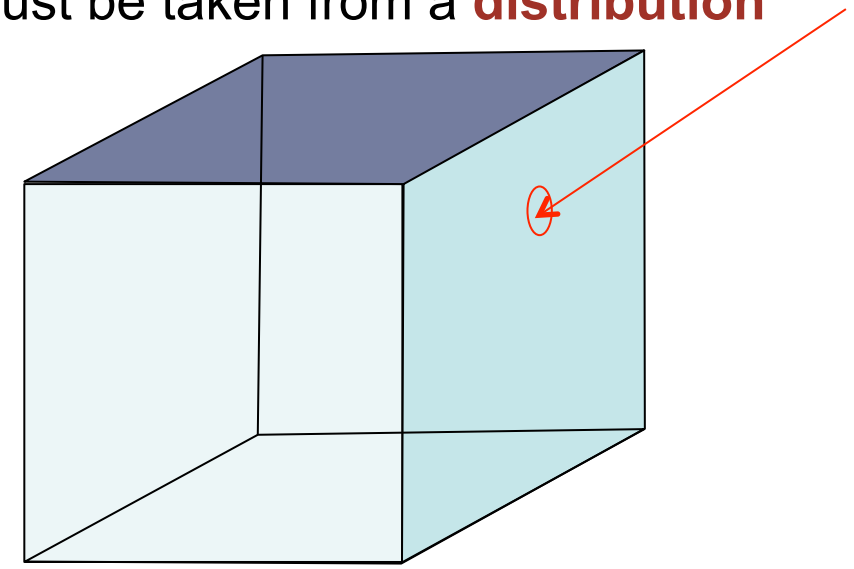
We start with a photon which has a direction according to the 3 directional cosines

u in direction x,
v in direction y,
w in direction z

and which is entering a volume (voxel) at x_0, y_0, z_0.

Individual particle tracking within the Monte Carlo method

Step 1: The track lenght d to the next interaction of an individual photon
– starting from the entry point –
can be anywhere.
For an individual photon it must be taken from a **distribution** determined by the
mean free path length d_{mfp}



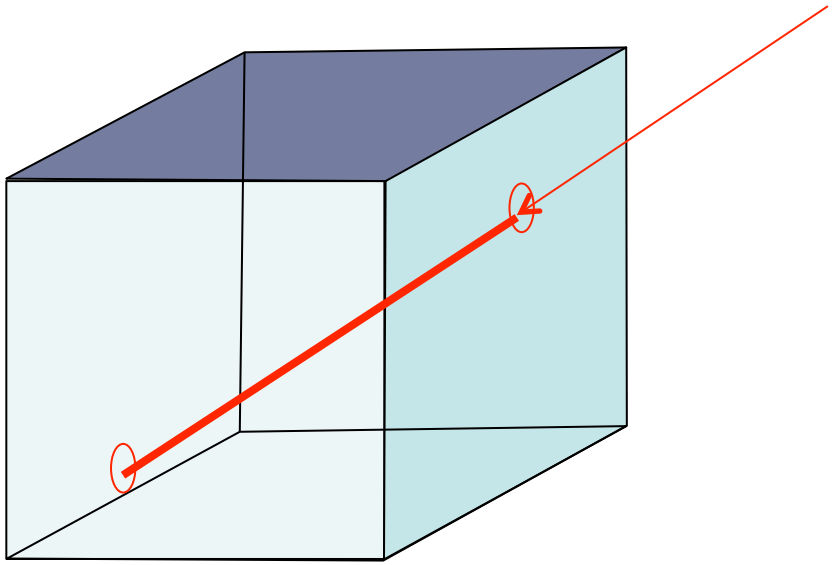
This is accomplished by a very simple method:

$$d_{sample} = -d_{mfp} \cdot \ln(r)$$

d_{sample} = distance to the next interaction for this **individual photon**
d_{mfp} = distance to the next interaction **on average**
r = random number out of the interval {0,1}

Individual particle tracking within the Monte Carlo method

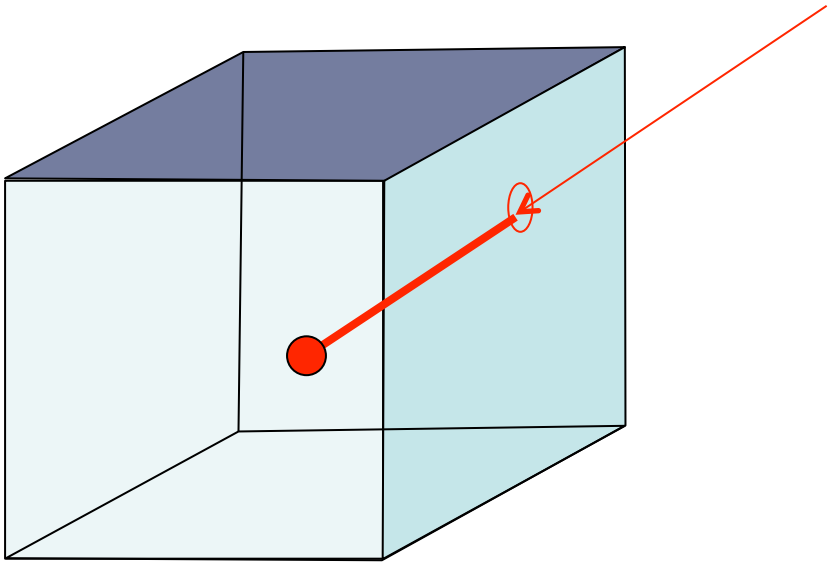
Step 2:
Also calculate the geometrical path length d_{geo} within V



Step 3: Make a differentiation between

Case 1: $d_{sample} < d_{geo}$

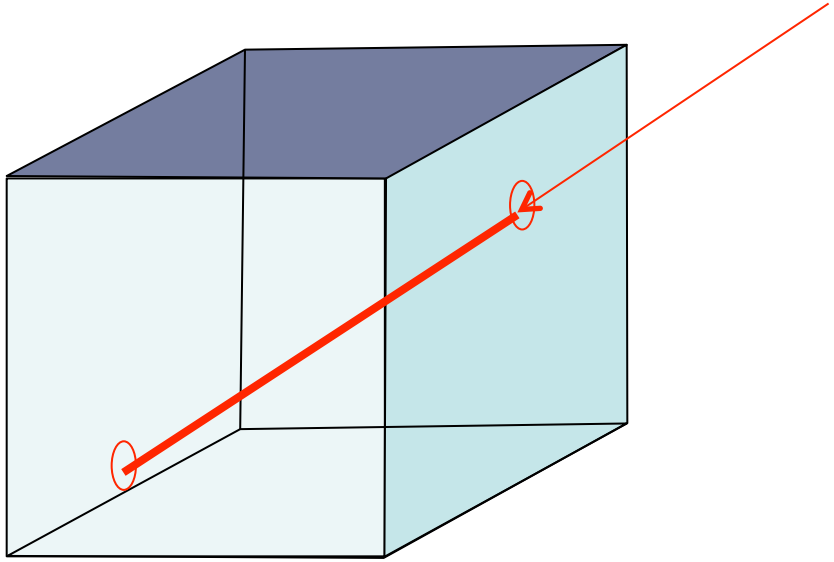
The interaction occurred within the voxel
Take d_{sample} for the track length



Case 2: $d_{sample} > d_{geo}$

No interaction within the voxel.

Take d_{geo} for the track length



Individual particle tracking within the Monte Carlo method

Step 4 in case that an interaction occured:

Determine energy and direction of the new photon (if produced) and continue tracking, now starting at the point of interaction

Step 4 in case that no interaction occured:

Go to adjacent voxel and determine the next $d_{sample,next}$ as:

$$d_{sample,next} = d_{sample} - d_{geo}$$

Step 5:
Repeat everything for any voxel and any new photon

Tracking in Monte Carlo Codes

More generally speaking, the term **tracking** can be used to describe the procedure of subsequently determining the trajectories in the **six dimensional phase space** between each two interactions.

The six dimensions are $(x;\Omega;E)$
where:

- $x = (x_1; x_2; x_3)$ are the spatial coordinate variable,
- Ω is the particle direction which is a point on a unit sphere S with the angles coordinates φ and θ
- E is the energy variable.

Summary: Treament Planning Systems

1) Computerized treatment planning is a part (however, an important part) within clinical treatment planning which consists of an entire chain of many steps:
2) Dose calculation again is a apart only within the treatment planning system.
3) Main methods of calculations are:
   ray tracing through a voxel geometry
   superposition using different kernel types
   tracking and energy scoring using MC
4) One should al least know the characteristics of a certain dose calculation method with respect to the requirement of an individual patient.