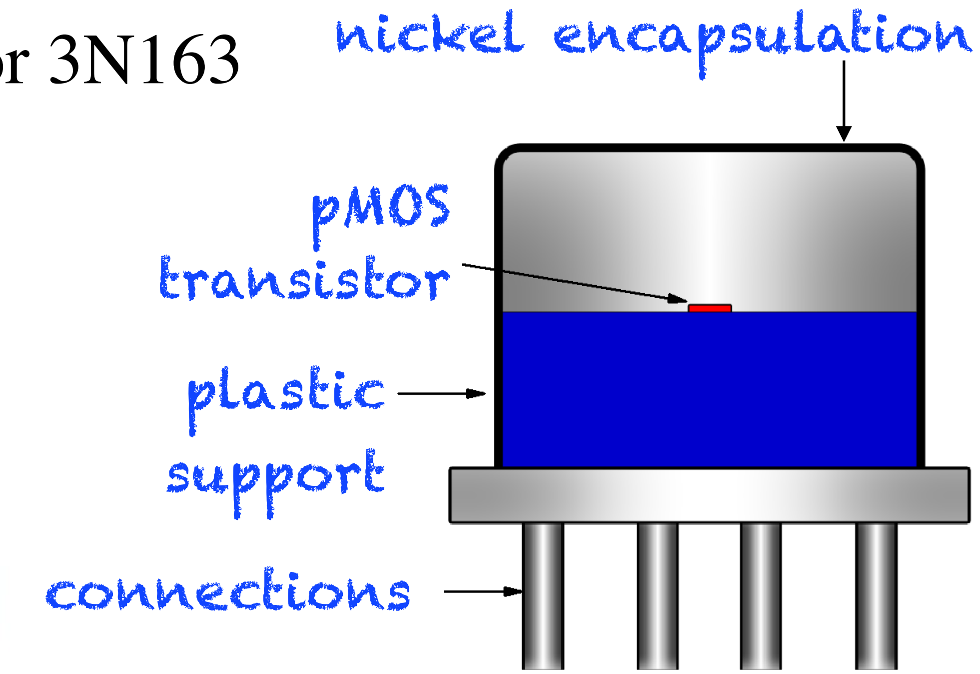
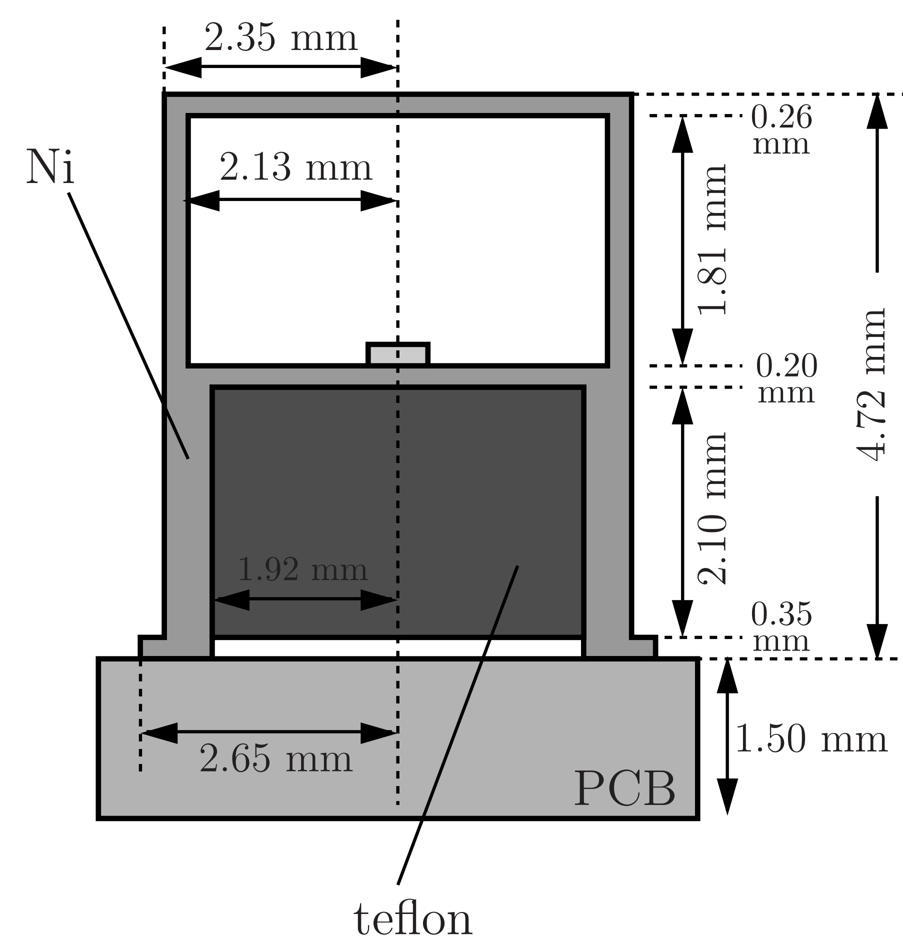
pMOS transistor 3N163

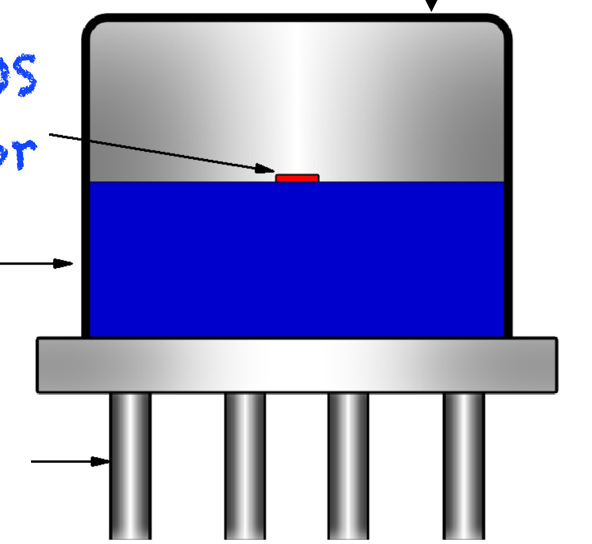
pMOS transistor 3N163



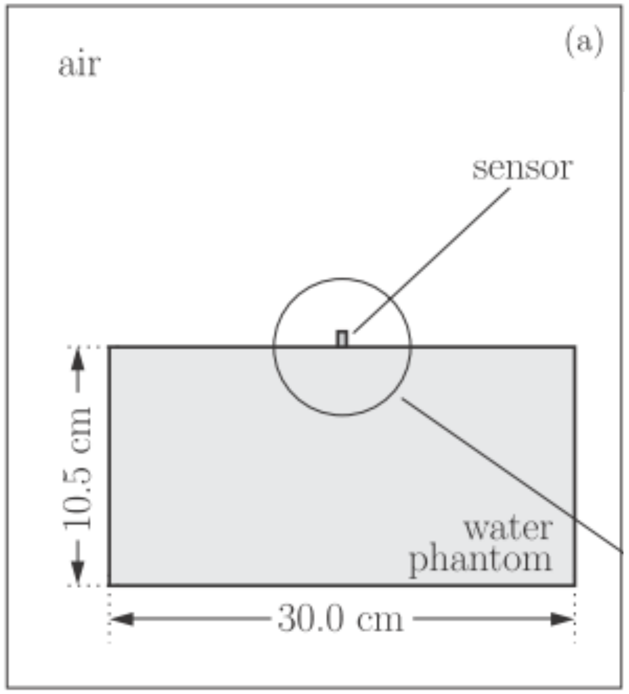


can this MOSFET be used as a dosimeter?

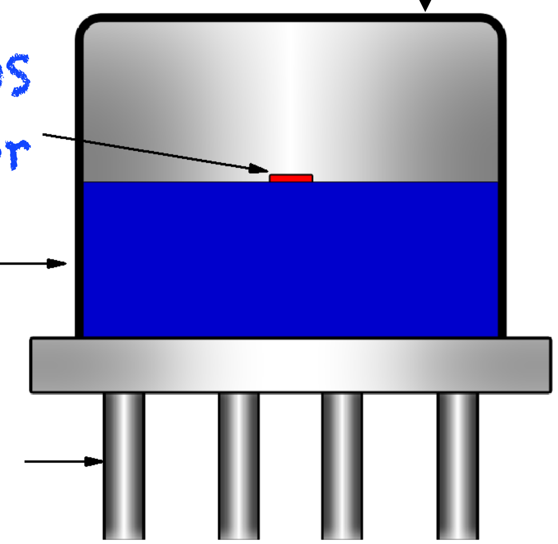
pMOS transistor 3N163



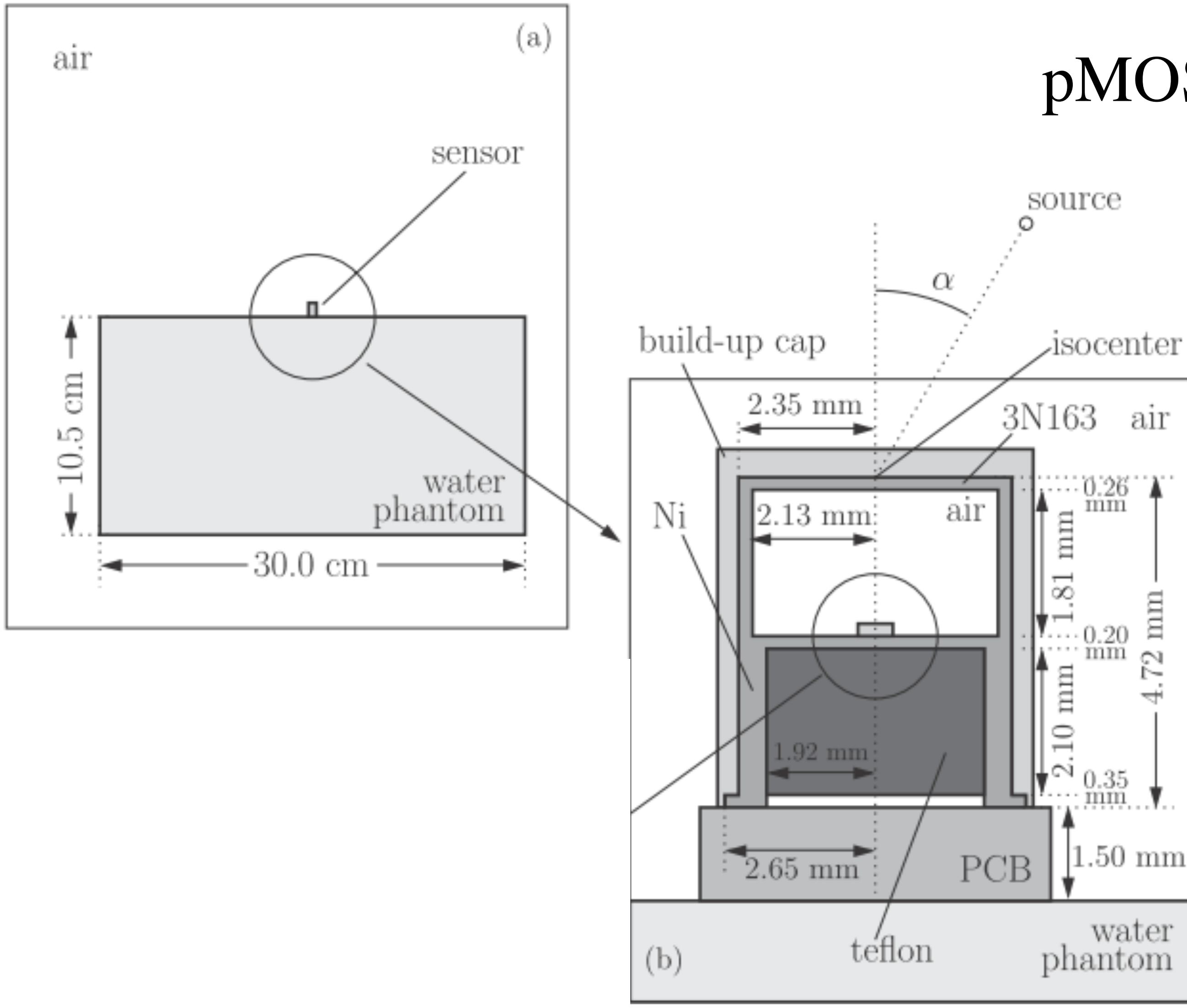
can this MOSFET be used as a dosimeter?



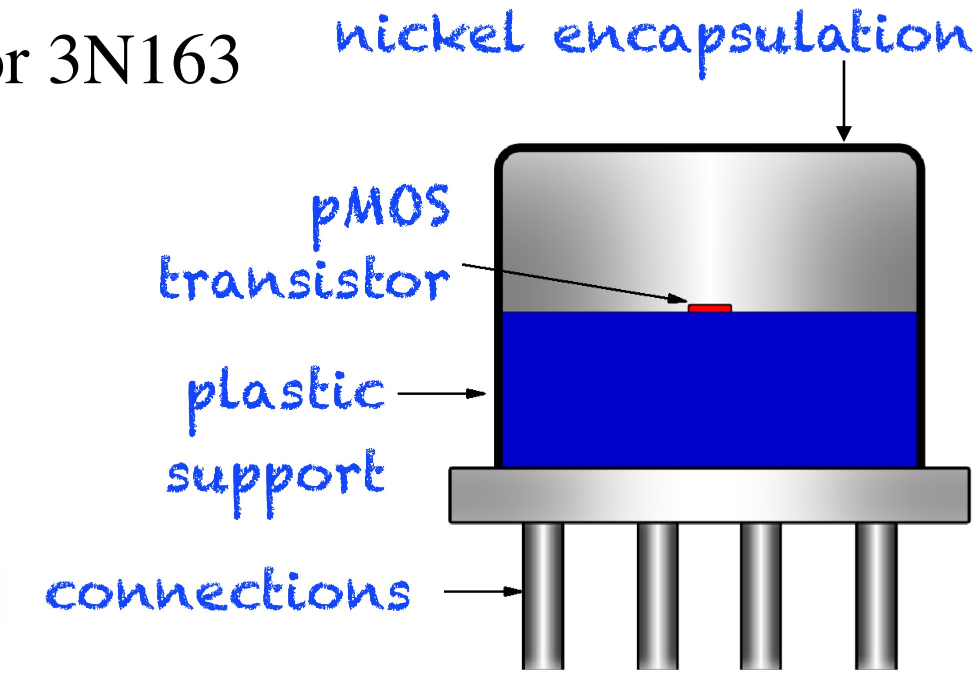
pMOS transistor 3N163



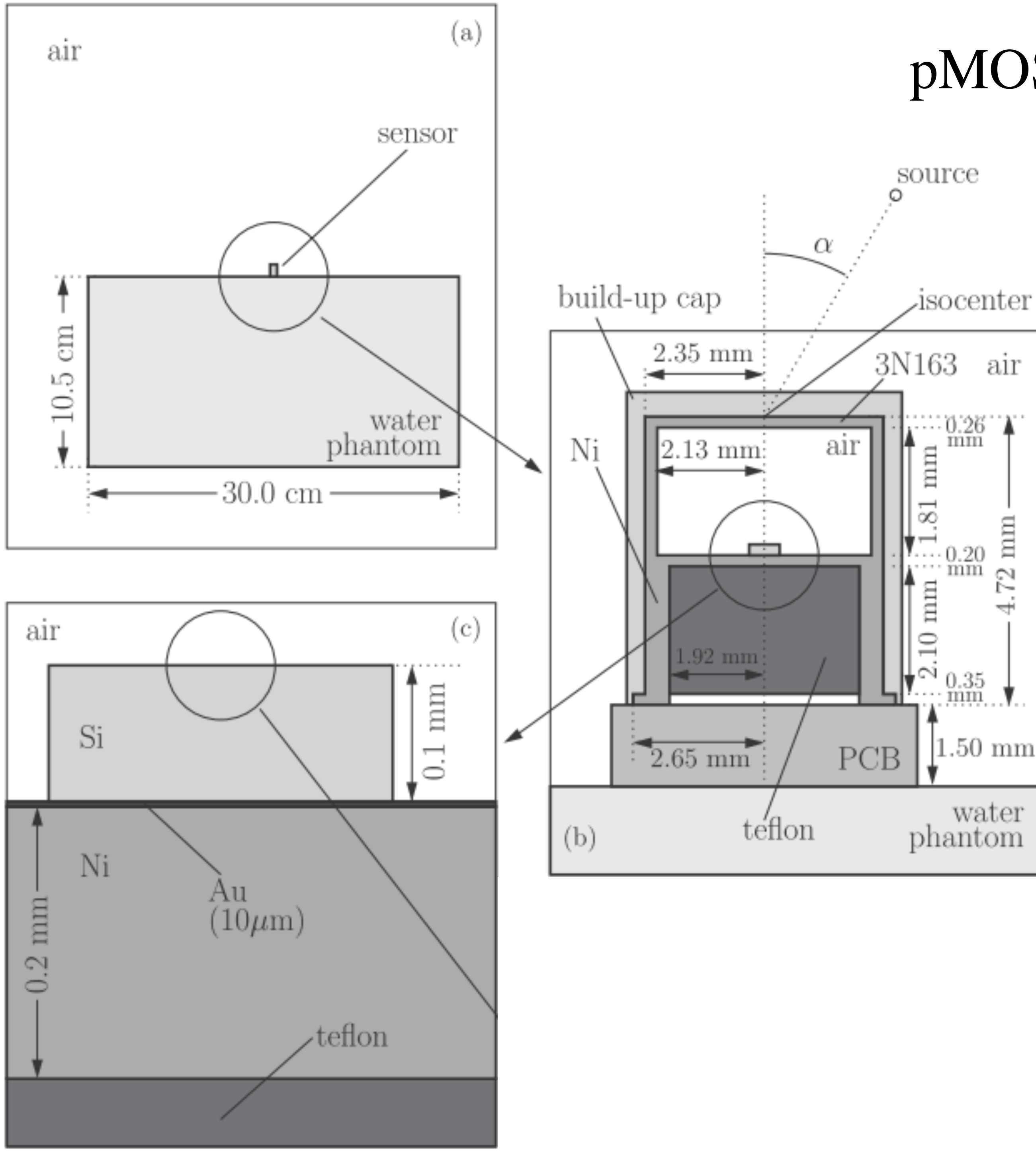
can this MOSFET be used as a dosimeter?



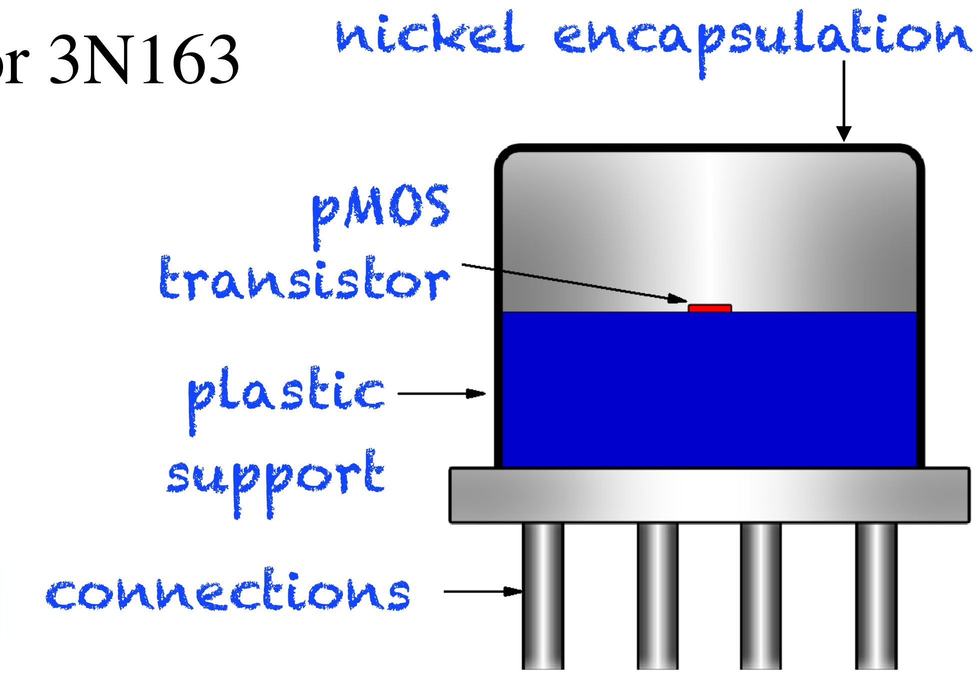
pMOS transistor 3N163



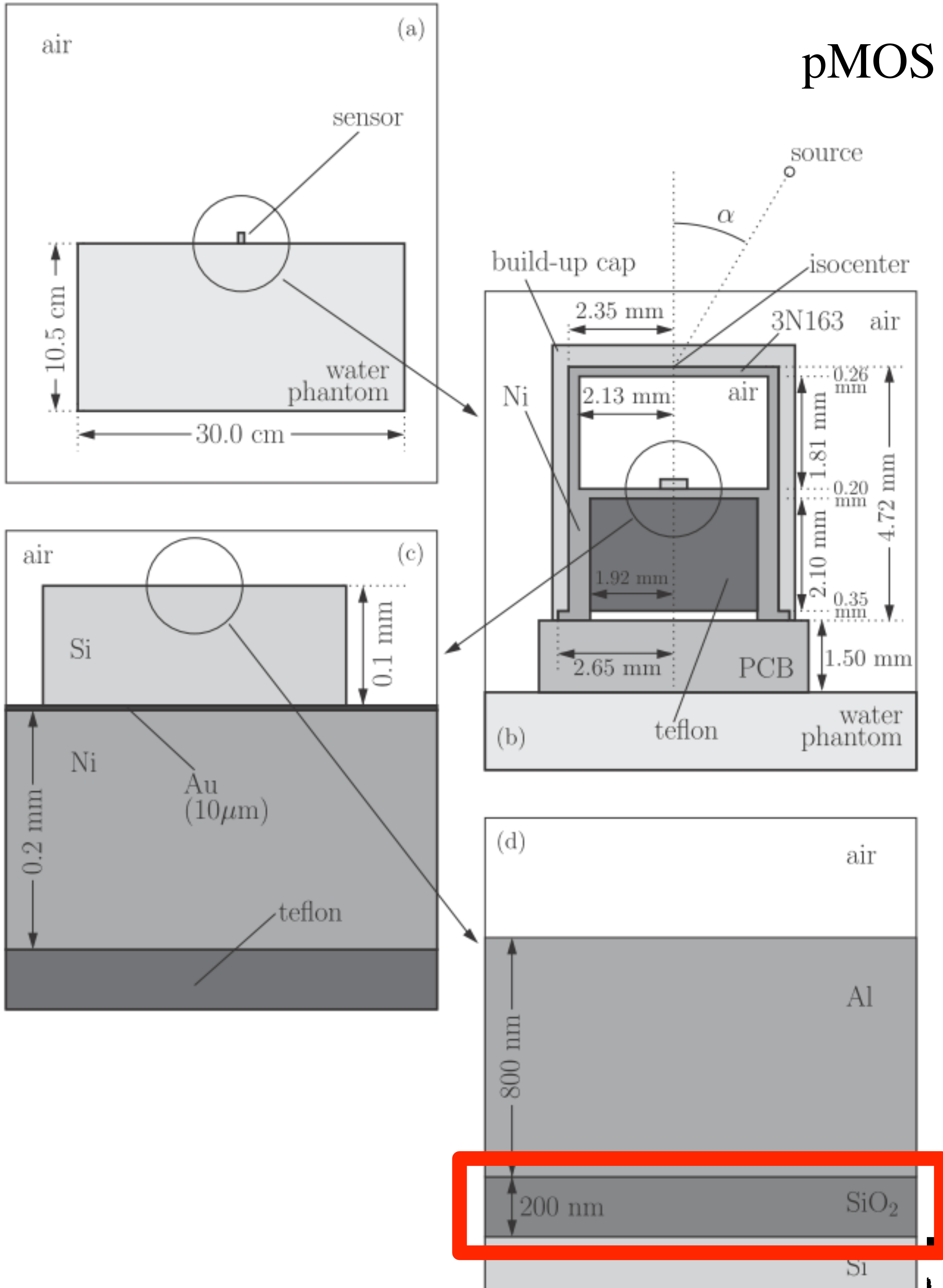
can this MOSFET be used as a dosimeter?



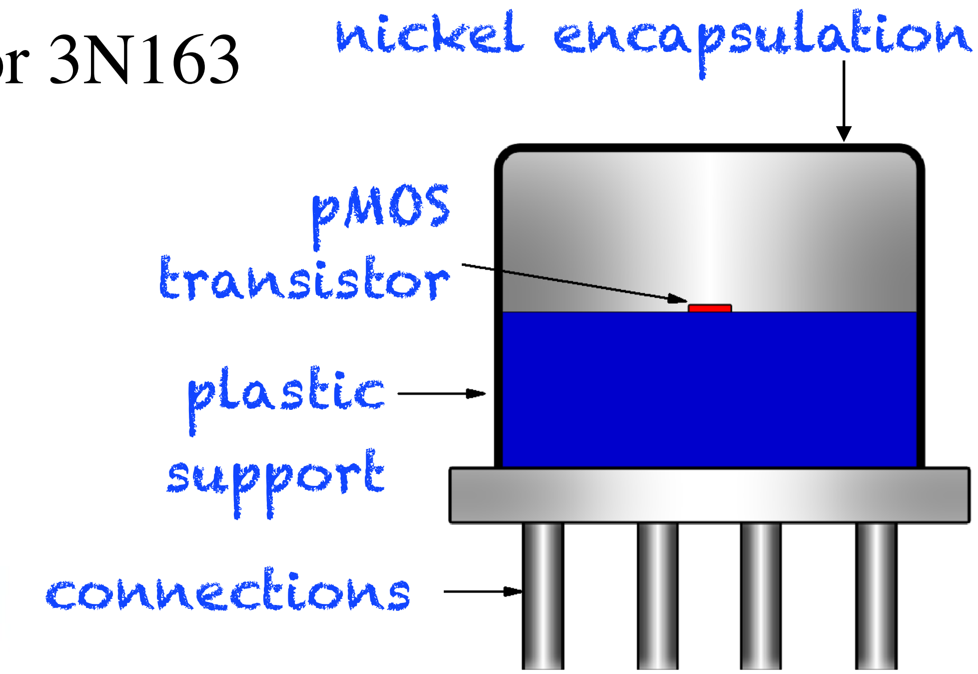
pMOS transistor 3N163



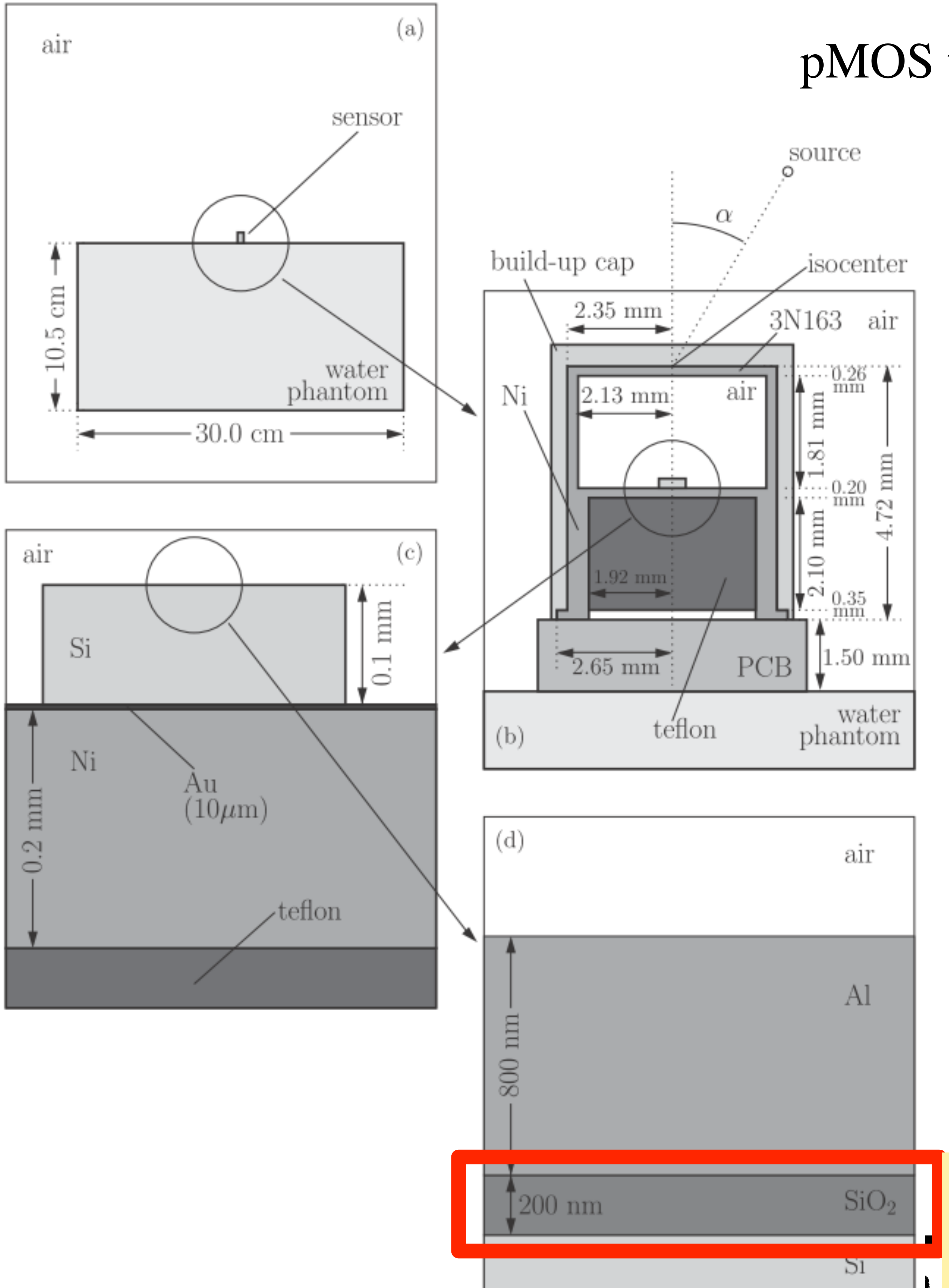
can this MOSFET be used as a dosimeter?



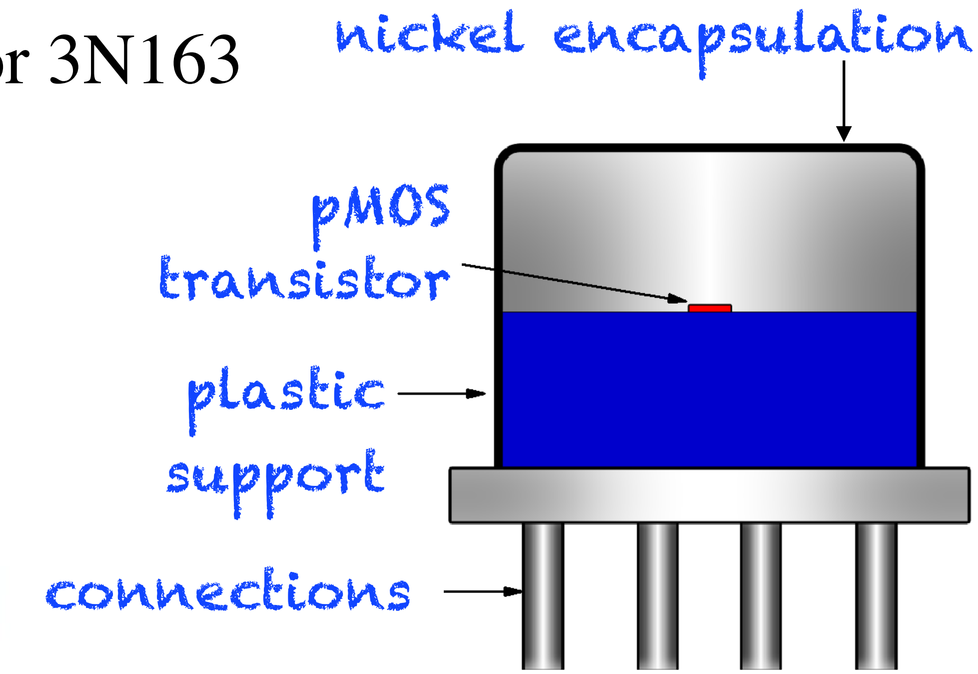
pMOS transistor 3N163



can this MOSFET be used as a dosimeter?

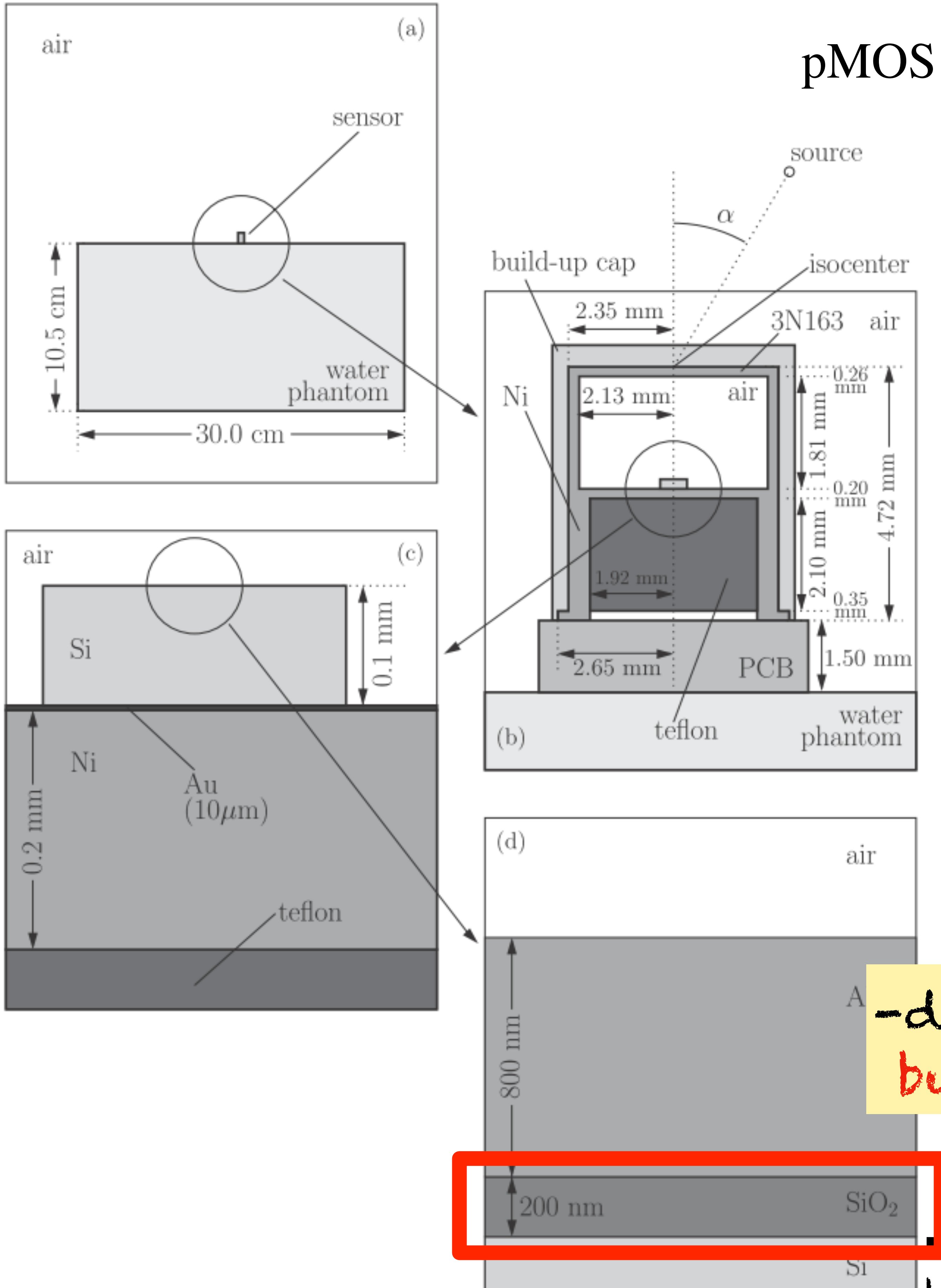


pMOS transistor 3N163

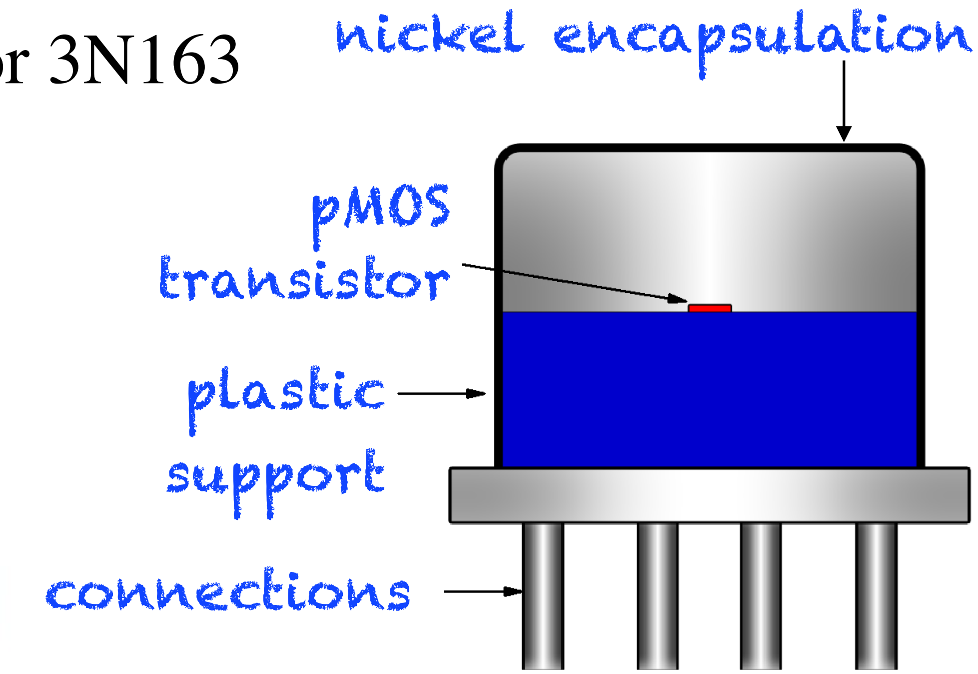


-MOSFET response:
energy deposited in the SiO_2 die

can this MOSFET be used as a dosimeter?



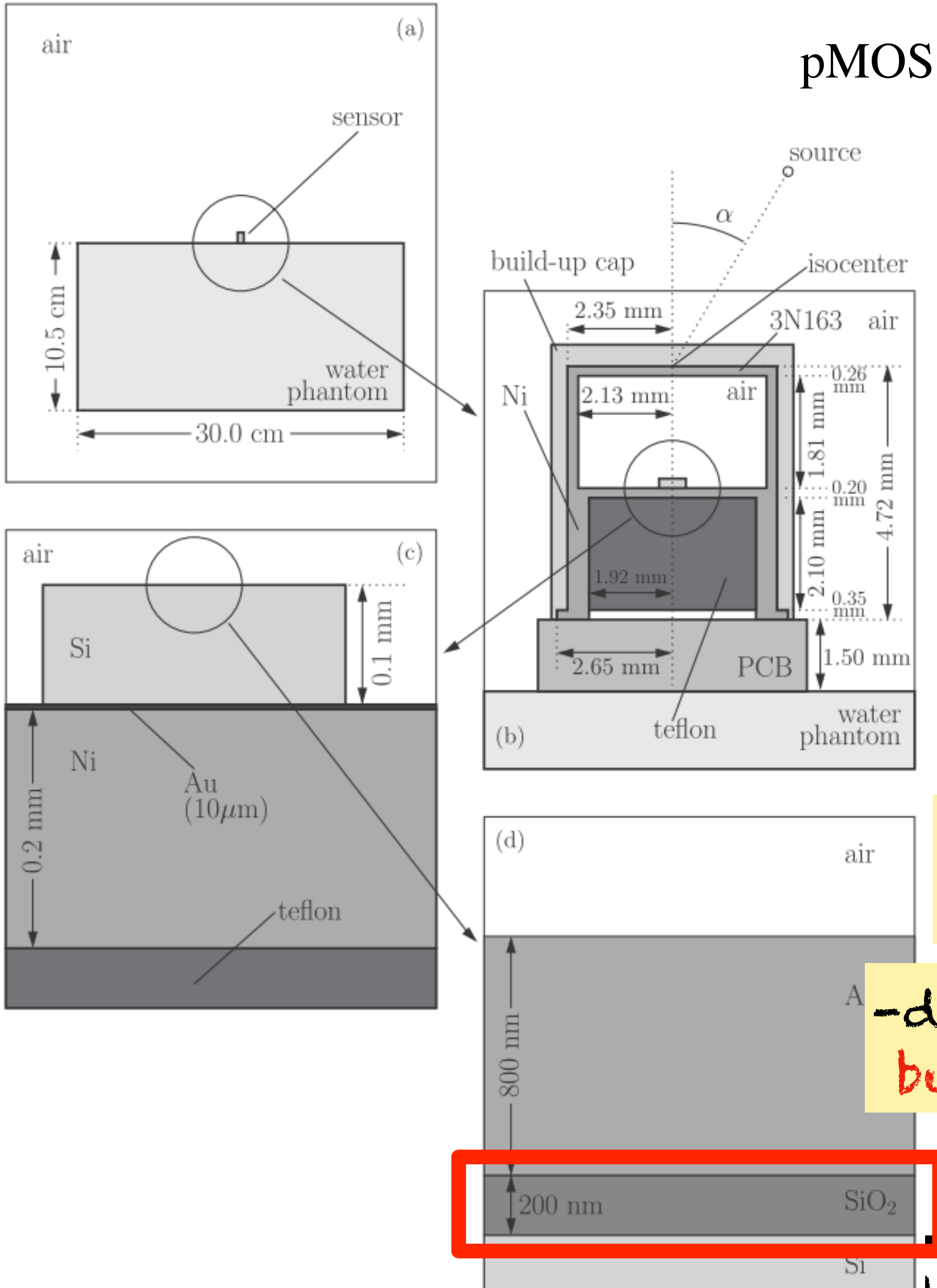
pMOS transistor 3N163



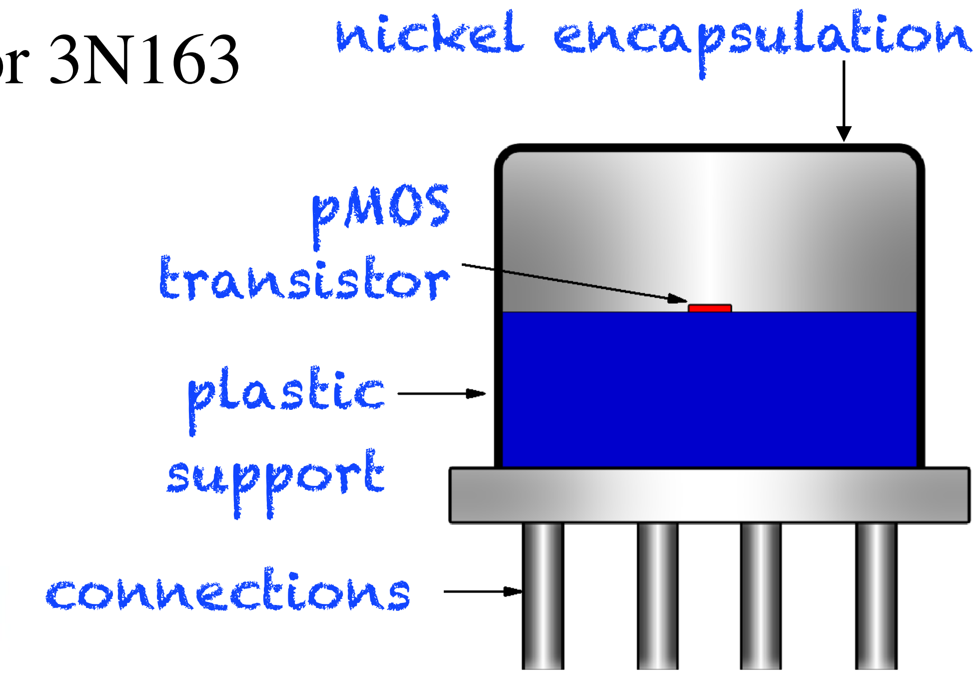
-detailed simulation within MOSFET,
but very low statistic

-MOSFET response:
energy deposited in the SiO_2 die

can this MOSFET be used as a dosimeter?



pMOS transistor 3N163

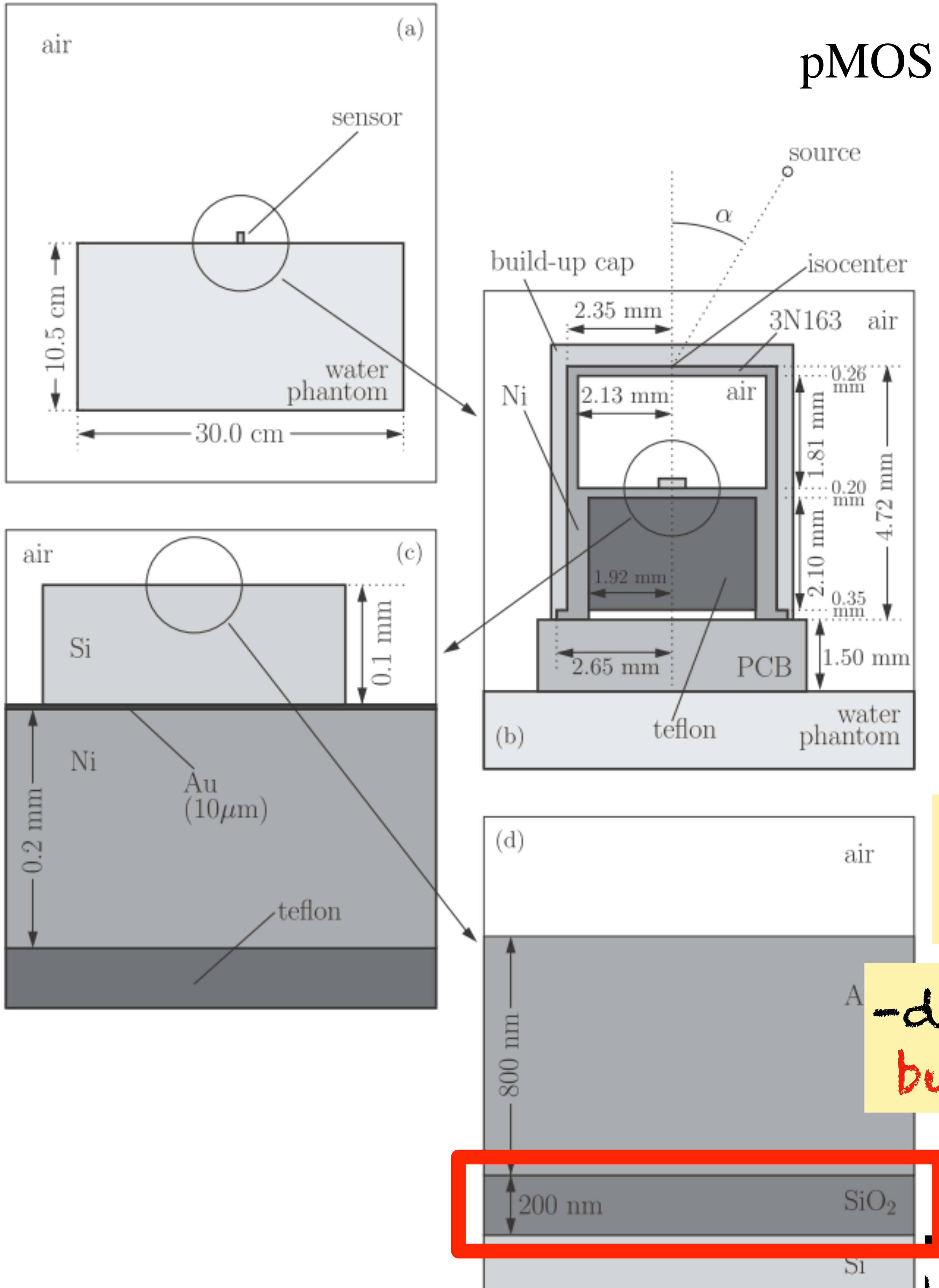


-huge computer cluster necessary:
brute force!!!

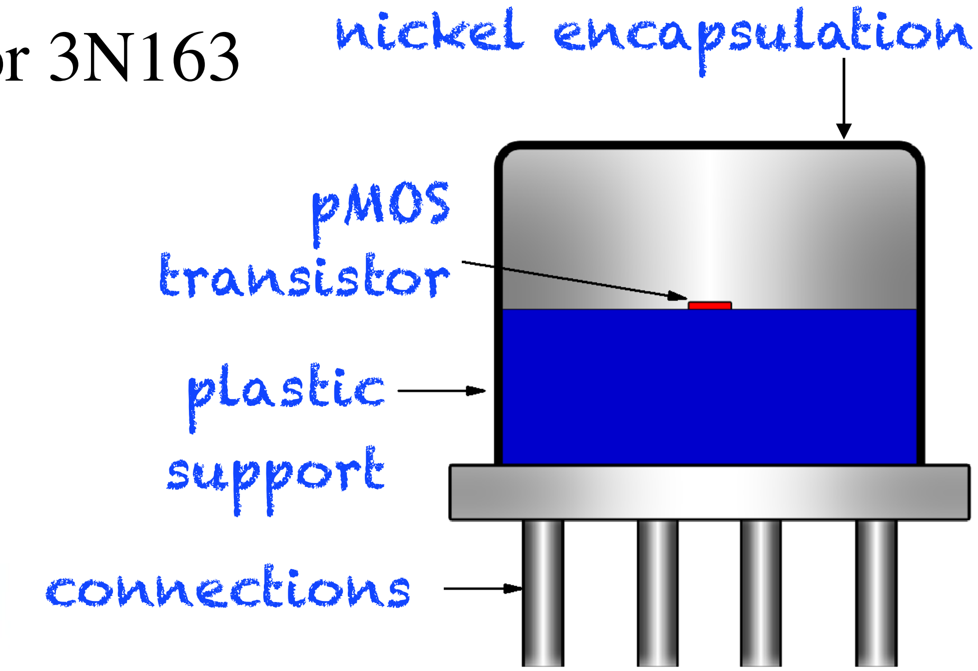
-detailed simulation within MOSFET,
but very low statistic

-MOSFET response:
energy deposited in the SiO_2 die

can this MOSFET be used as a dosimeter?



pMOS transistor 3N163



-or use
variance reduction techniques

-huge computer cluster necessary:
brute force!!!

-detailed simulation within MOSFET,
but very low statistic

-MOSFET response:
energy deposited in the SiO_2 die

Variance reduction techniques in Monte Carlo simulations: ants at work!

Salvador García-Pareja
Hospital Regional Universitario, Málaga, Spain
Antonio M. Lallena
Universidad de Granada, Spain

Hospital Regional Universitario de Málaga

UNIVERSIDAD DE GRANADA

Outline

- Introduction
- Ant colony algorithm
- Results
- Conclusions

Introduction

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

- in particular:
radiation transport in matter

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

- in particular:
radiation transport in matter

surface spectroscopy –
electron microscopy – microanalysis with electronic probes –
radiation detector design – dosimetry – radiotherapy – ...

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

- in particular:
radiation transport in matter

Monte Carlo tool: any mathematical procedure that uses random numbers to solve problems

surface spectroscopy -
electron microscopy - microanalysis with electronic probes -
radiation detector design - dosimetry - radiotherapy - ...

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

- in particular:
radiation transport in matter

Monte Carlo tool: any mathematical procedure that uses random numbers to solve problems

surface spectroscopy -
electron microscopy - microanalysis with electronic probes -
radiation detector design - dosimetry - radiotherapy - ...

- high-energy particles traveling through matter: suffer interactions, transfer energy and produce new particles

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

Monte Carlo tool: any mathematical procedure that uses random numbers to solve problems

- in particular:
radiation transport in matter

surface spectroscopy - electron microscopy - microanalysis with electronic probes - radiation detector design - dosimetry - radiotherapy - ...

- high-energy particles traveling through matter: suffer interactions, transfer energy and produce new particles

particle showers are formed

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

Monte Carlo tool: any mathematical procedure that uses random numbers to solve problems

- in particular:
radiation transport in matter

surface spectroscopy –
electron microscopy – microanalysis with electronic probes –
radiation detector design – dosimetry – radiotherapy – ...

- high-energy particles traveling through matter: suffer interactions, transfer energy and produce new particles

particle showers are formed

- Monte Carlo simulation: produces exact results

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

Monte Carlo tool: any mathematical procedure that uses random numbers to solve problems

- in particular:
radiation transport in matter

surface spectroscopy -
electron microscopy - microanalysis with electronic probes -
radiation detector design - dosimetry - radiotherapy - ...

- high-energy particles traveling through matter: suffer interactions, transfer energy and produce new particles

particle showers are formed

- Monte Carlo simulation:
produces exact results

but: statistical uncertainties!!

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

Monte Carlo tool: any mathematical procedure that uses random numbers to solve problems

- in particular:
radiation transport in matter

surface spectroscopy -
electron microscopy - microanalysis with electronic probes -
radiation detector design - dosimetry - radiotherapy - ...

- high-energy particles traveling through matter: suffer interactions, transfer energy and produce new particles

particle showers are formed

- Monte Carlo simulation:
produces exact results

but: statistical uncertainties!!

- better precision and accuracy requires increasing the number of simulated histories

Introduction

- increase of the computer power has made Monte Carlo simulation a powerful tool in many fields

Monte Carlo tool: any mathematical procedure that uses random numbers to solve problems

- in particular:
radiation transport in matter

surface spectroscopy -
electron microscopy - microanalysis with electronic probes -
radiation detector design - dosimetry - radiotherapy - ...

- high-energy particles traveling through matter: suffer interactions, transfer energy and produce new particles

particle showers are formed

- Monte Carlo simulation:
produces exact results

but: statistical uncertainties!!

large calculation CPU times!!

- better precision and accuracy requires increasing the number of simulated histories

variance reduction techniques

variance reduction techniques

procedures permitting the reduction of the statistical uncertainties without increasing the calculation time

variance reduction techniques

procedures permitting the reduction of the statistical uncertainties without increasing the calculation time

Obvious VRTs:
-avoid useless calculations
-take advantage of the problem symmetries

variance reduction techniques

procedures permitting the reduction of the statistical uncertainties without increasing the calculation time

Obvious VRTs:
- avoid useless calculations
- take advantage of the problem symmetries

- Russian roulette
- splitting
- interaction forcing
- directional bremsstrahlung splitting
- range rejection

variance reduction techniques

procedures permitting the reduction of the statistical uncertainties without increasing the calculation time

Obvious VRTs:
- avoid useless calculations
- take advantage of the problem symmetries

- Russian roulette
- splitting
- interaction forcing
- directional bremsstrahlung splitting
- range rejection

- statistical weight: unbiased simulations!!

Introduction

variance reduction techniques

procedures permitting the reduction of the statistical uncertainties without increasing the calculation time

Obvious VRTs:
- avoid useless calculations
- take advantage of the problem symmetries

- Russian roulette
- splitting
- interaction forcing
- directional bremsstrahlung splitting
- range rejection

- statistical weight: unbiased simulations!!

- VRTs used properly increase the efficiency of the simulation!!

variance reduction techniques

procedures permitting the reduction of the statistical uncertainties without increasing the calculation time

Obvious VRTs:
-avoid useless calculations
-take advantage of the problem symmetries

- Russian roulette
- splitting
- interaction forcing
- directional bremsstrahlung splitting
- range rejection

- statistical weight: unbiased simulations!!

- VRTs used properly increase the efficiency of the simulation!!

how to do it?

Introduction

- splitting

- Russian roulette

Introduction

- **splitting**

a particle in a state $(\mathbf{r}, E, \hat{\mathbf{d}})$,
with weight w_0, moving towards a RoI,
is transformed in $\mathcal{S} > 1$ particles,
in the same state, with weight $w = w_0/\mathcal{S}$

- **Russian roulette**

• splitting

a particle in a state $(\mathbf{r}, E, \hat{\mathbf{d}})$,
with weight w_0, moving towards a RoI,
is transformed in $\mathcal{S} > 1$ particles,
in the same state, with weight $w = w_0/\mathcal{S}$

• Russian roulette

a particle with weight w_0,
moving away from the RoI,
is "killed" with probability $\mathcal{K}\,(0 < \mathcal{K} < 1)$;
if it survives, its weight becomes $w = w_0/(1 - \mathcal{K})$

Introduction

• splitting

a particle in a state $(\mathbf{r}, E, \hat{\mathbf{d}})$,
with weight w_0, moving towards a RoI,
is transformed in $\mathcal{S} > 1$ particles,
in the same state, with weight $w = w_0/\mathcal{S}$

• Russian roulette

a particle with weight w_0,
moving away from the RoI,
is "killed" with probability $\mathcal{K}\,(0 < \mathcal{K} < 1)$;
if it survives, its weight becomes $w = w_0/(1 - \mathcal{K})$

• splitting:
reduces variance but
may increase CPU time

• Russian roulette:
increases variance but
reduces CPU time

Introduction

- **splitting**

a particle in a state $(\mathbf{r}, E, \hat{\mathbf{d}})$,
with weight w_0, moving towards a RoI,
is transformed in $\mathcal{S} > 1$ particles,
in the same state, with weight $w = w_0/\mathcal{S}$

- **Russian roulette**

a particle with weight w_0,
moving away from the RoI,
is "killed" with probability $\mathcal{K}\,(0 < \mathcal{K} < 1)$;
if it survives, its weight becomes $w = w_0/(1 - \mathcal{K})$

➡usually used together

- splitting:
reduces variance but
may increase CPU time

- Russian roulette:
increases variance but
reduces CPU time

Introduction

- **splitting**

a particle in a state $(\mathbf{r}, E, \hat{\mathbf{d}})$,
with weight w_0, moving towards a RoI,
is transformed in $\mathcal{S} > 1$ particles,
in the same state, with weight $w = w_0/\mathcal{S}$

- **Russian roulette**

a particle with weight w_0,
moving away from the RoI,
is "killed" with probability $\mathcal{K}$ $(0 < \mathcal{K} < 1)$;
if it survives, its weight becomes $w = w_0/(1 - \mathcal{K})$

➡ usually used together

➡ in problems with a well defined RoI

- splitting:
  reduces variance but
  may increase CPU time

- Russian roulette:
  increases variance but
  reduces CPU time

• **splitting**

a particle in a state $(\mathbf{r}, E, \hat{\mathbf{d}})$,
with weight w_0, moving towards a RoI,
is transformed in $\mathcal{S} > 1$ particles,
in the same state, with weight $w = w_0/\mathcal{S}$

• **Russian roulette**

a particle with weight w_0,
moving away from the RoI,
is "killed" with probability $\mathcal{K}$ $(0 < \mathcal{K} < 1)$;
if it survives, its weight becomes $w = w_0/(1 - \mathcal{K})$

➡ usually used together

➡ in problems with a well defined RoI

➡ favor radiation flux toward the RoI and inhibit radiation moving away from it: used together save time computing wasted to track particles not contributing to the quantities of interest scored within the RoI

• splitting:
reduces variance but may increase CPU time

• Russian roulette:
increases variance but reduces CPU time

Introduction

• splitting

a particle in a state $(\mathbf{r}, E, \hat{\mathbf{d}})$,
with weight w_0, moving towards a RoI,
is transformed in $\mathcal{S} > 1$ particles,
in the same state, with weight $w = w_0/\mathcal{S}$

• Russian roulette

a particle with weight w_0,
moving away from the RoI,
is "killed" with probability $\mathcal{K}$ $(0 < \mathcal{K} < 1)$;
if it survives, its weight becomes $w = w_0/(1 - \mathcal{K})$

➡ usually used together

➡ in problems with a well defined RoI

➡ favor radiation flux toward the RoI and inhibit radiation moving away from it: used together save time computing wasted to track particles not contributing to the quantities of interest scored within the RoI

• splitting:
reduces variance but may increase CPU time

• Russian roulette:
increases variance but reduces CPU time

VRT effectiveness depends on:

- $\mathcal{S}$ and $\mathcal{K}$ values
- strategy used for splitting and killing

ant colony algorithm: Implementation

ant colony algorithm: Implementation

-ants look for food following random walks
-if food is found, ants come back to the nest depositing pheromone
-ants tend to follow paths with the largest level of pheromone
-the level of pheromone increases in the optimal paths between nest and food

ant colony algorithm: Implementation

-ants look for food following random walks
-if food is found, ants come back to the nest depositing pheromone
-ants tend to follow paths with the largest level of pheromone
-the level of pheromone increases in the optimal paths between nest and food

particle source

ant colony algorithm: Implementation

-ants look for food following random walks
-if food is found, ants come back to the nest depositing pheromone
-ants tend to follow paths with the largest level of pheromone
-the level of pheromone increases in the optimal paths between nest and food

particle source

particles (photons)

ant colony algorithm: Implementation

-ants look for food following random walks
-if food is found, ants come back to the nest depositing pheromone
-ants tend to follow paths with the largest level of pheromone
-the level of pheromone increases in the optimal paths between nest and food

particle source

particle trajectories

particles (photons)

ant colony algorithm: Implementation

-ants look for food following random walks
-if food is found, ants come back to the nest depositing pheromone
-ants tend to follow paths with the largest level of pheromone
-the level of pheromone increases in the optimal paths between nest and food

particle source

particle trajectories

RoI

particles (photons)

ant colony algorithm: Implementation

- ants look for food following random walks
- if food is found, ants come back to the nest depositing pheromone
- ants tend to follow paths with the largest level of pheromone
- the level of pheromone increases in the optimal paths between nest and food

particle source

particle trajectories

RoI

importance

particles (photons)

ant colony algorithm: Implementation

-ants look for food following random walks
-if food is found, ants come back to the nest depositing pheromone
-ants tend to follow paths with the largest level of pheromone
-the level of pheromone increases in the optimal paths between nest and food

nest → particle source

paths → particle trajectories

food → RoI

ants → particles (photons)

pheromone → importance

-a particle source and a RoI are defined
-the whole geometry is divided into virtual cells
-each cell is characterized by an importance value

-importance in a cell tells about the probability that a particle passing through it reaches the ROI

-importance in a cell tells about the probability that a particle passing through it reaches the ROI

$$P_i = \frac{\mathcal{W}_i^{\mathrm{C}}}{\mathcal{W}_i^{\mathrm{P}}}$$

sum of the weights of particles entering i-th cell and that reach the RoI (they or their descendants)

sum of the weights of particles entering i-th cell

-importance in a cell tells about the probability that a particle passing through it reaches the ROI

$$P_i = \frac{\mathcal{W}_i^{\mathrm{C}}}{\mathcal{W}_i^{\mathrm{P}}}$$

$\mathcal{W}_i^{\mathrm{C}}$: sum of the weights of particles entering i-th cell and that reach the RoI (they or their descendants)

$\mathcal{W}_i^{\mathrm{P}}$: sum of the weights of particles entering i-th cell

importance

$$I = 2^k$$

$$k = \begin{cases} \left[5\frac{P(i)}{P(0)} - 5\right], & \text{if } P(i) \le P(0)\,, \\ \left[7\frac{P(i) - P(0)}{1 - P(0)}\right], & \text{if } P(i) > P(0)\,. \end{cases}$$

-importance in a cell tells about the probability that a particle passing through it reaches the ROI

$$P_i = \frac{\mathcal{W}_i^{\mathrm{C}}}{\mathcal{W}_i^{\mathrm{P}}}$$

sum of the weights of particles entering i-th cell and that reach the RoI (they or their descendants)

sum of the weights of particles entering i-th cell

importance

$$I = 2^k$$

$$k = \begin{cases} \left[5\frac{P(i)}{P(0)} - 5\right], & \text{if } P(i) \leq P(0)\,, \\ \left[7\frac{P(i) - P(0)}{1 - P(0)}\right], & \text{if } P(i) > P(0)\,. \end{cases}$$

-once a particle enters a new cell, VRTs are applied according the particle weight (w) and the cell importance (I)

-importance in a cell tells about the probability that a particle passing through it reaches the ROI

$$P_i = \frac{\mathcal{W}_i^{\mathrm{C}}}{\mathcal{W}_i^{\mathrm{P}}}$$

$\mathcal{W}_i^{\mathrm{C}}$: sum of the weights of particles entering i-th cell and that reach the RoI (they or their descendants)

$\mathcal{W}_i^{\mathrm{P}}$: sum of the weights of particles entering i-th cell

$$I = 2^k$$

importance

$$k = \begin{cases} \left[5\dfrac{P(i)}{P(0)} - 5\right], & \text{if } P(i) \le P(0), \\ \left[7\dfrac{P(i) - P(0)}{1 - P(0)}\right], & \text{if } P(i) > P(0). \end{cases}$$

-once a particle enters a new cell, VRTs are applied according the particle weight (w) and the cell importance (I)

- if w•I>1: splitting in w•I particles with w'=1/I

-importance in a cell tells about the probability that a particle passing through it reaches the ROI

$$P_i = \frac{\mathcal{W}_i^{\mathrm{C}}}{\mathcal{W}_i^{\mathrm{P}}}$$

sum of the weights of particles entering i-th cell and that reach the RoI (they or their descendants)

sum of the weights of particles entering i-th cell

importance

$$I = 2^k$$

$$k = \begin{cases} \left[5\dfrac{P(i)}{P(0)} - 5\right], & \text{if } P(i) \leq P(0)\,, \\ \left[7\dfrac{P(i) - P(0)}{1 - P(0)}\right], & \text{if } P(i) > P(0)\,. \end{cases}$$

-once a particle enters a new cell, VRTs are applied according the particle weight (w) and the cell importance (I)

- if w·I>1: splitting in w·I particles with w'=1/I
- if w·I<1: apply Rr with survival probability w·I;
  if particles survives: w'=1/I

-importance in a cell tells about the probability that a particle passing through it reaches the ROI

$$P_i = \frac{\mathcal{W}_i^{\mathrm{C}}}{\mathcal{W}_i^{\mathrm{P}}}$$

$\mathcal{W}_i^{\mathrm{C}}$: sum of the weights of particles entering i-th cell and that reach the RoI (they or their descendants)

$\mathcal{W}_i^{\mathrm{P}}$: sum of the weights of particles entering i-th cell

importance

$$I = 2^k$$

$$k = \begin{cases} \left[5\dfrac{P(i)}{P(0)} - 5\right], & \text{if } P(i) \leq P(0)\,, \\ \left[7\dfrac{P(i) - P(0)}{1 - P(0)}\right], & \text{if } P(i) > P(0)\,. \end{cases}$$

-once a particle enters a new cell, VRTs are applied according the particle weight (w) and the cell importance (I)

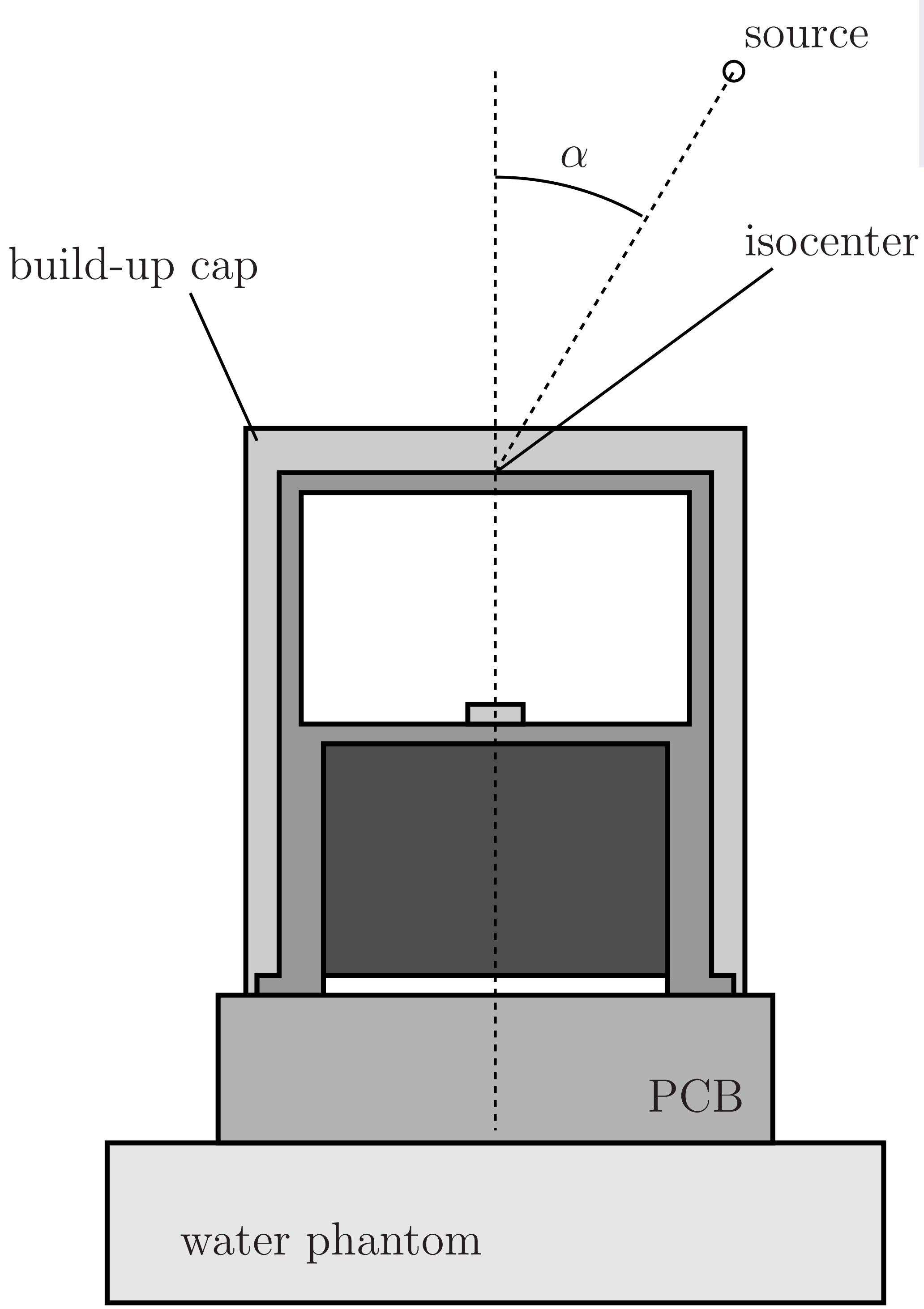
- if w·I>1: splitting in w·I particles with w'=1/I
- if w·I<1: apply Rr with survival probability w·I; if particles survives: w'=1/I
- if w·I=1: do nothing

Results

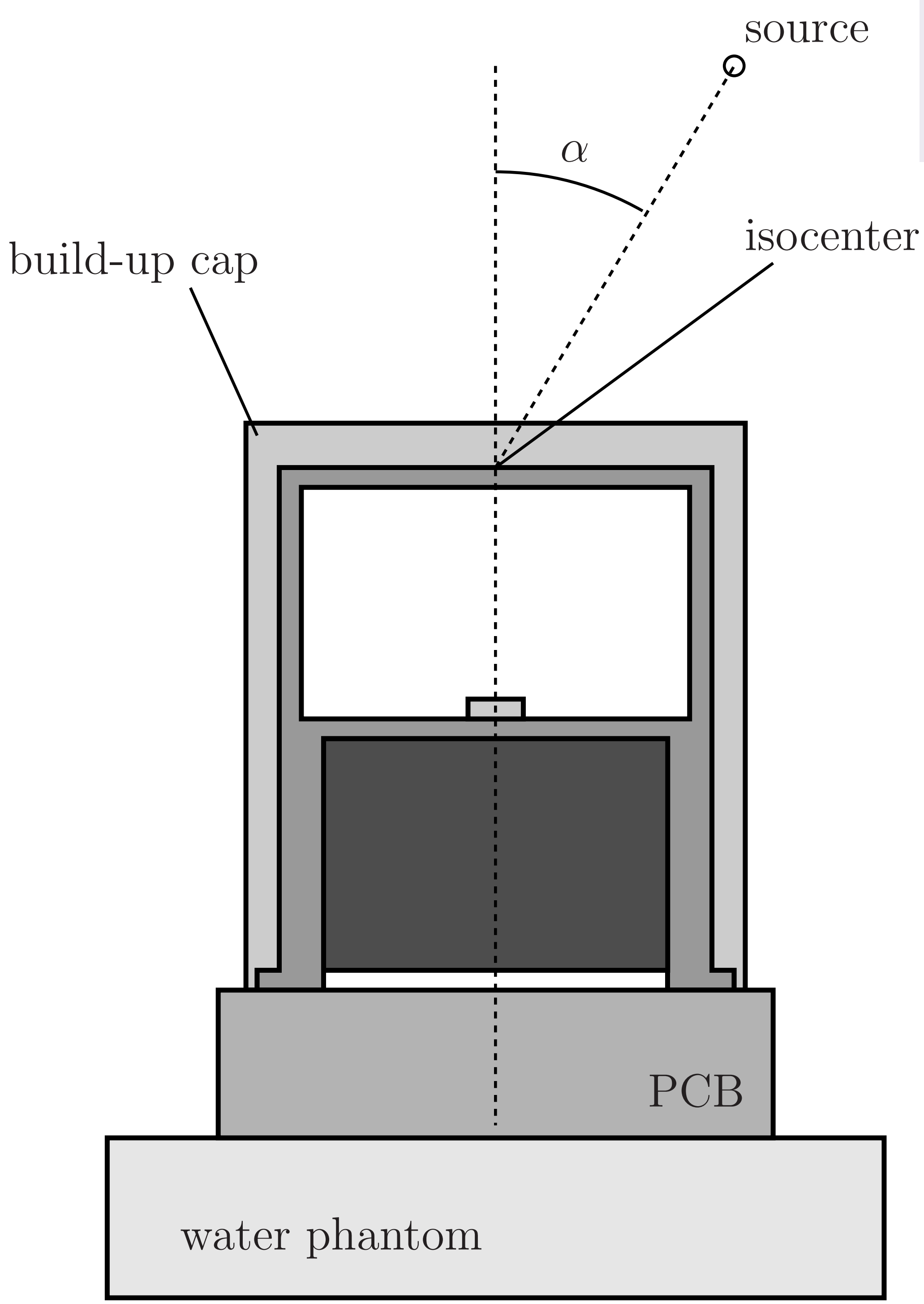
MOSFET used as a in-vivo dosimeter

MOSFET used as a in-vivo dosimeter

-address the possible angular dependence of the MOSFET response to irradiation



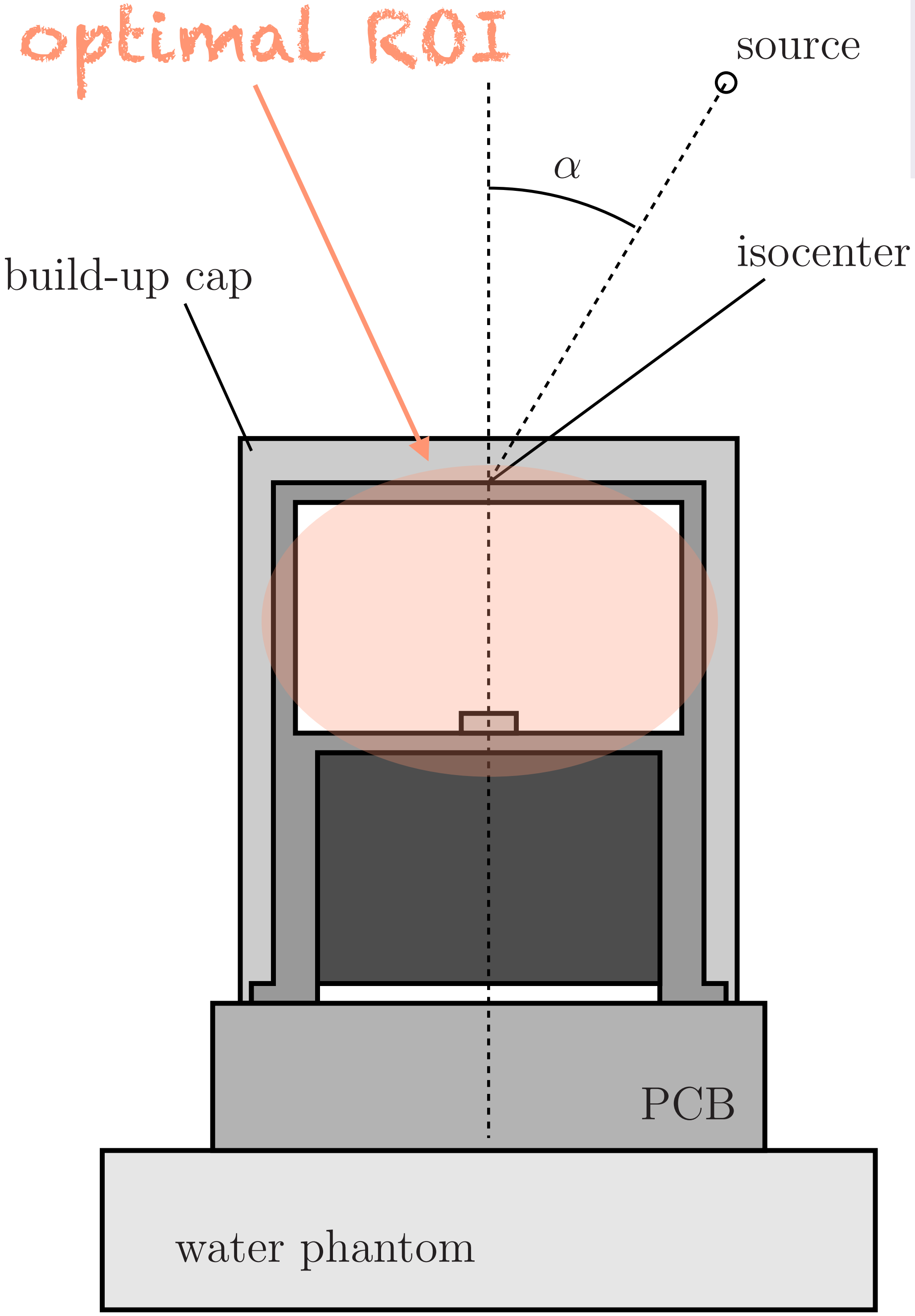
MOSFET used as a in-vivo dosimeter



-address the possible angular dependence of the MOSFET response to irradiation

- If RoI is restricted to SiO_2
  -the algorithm does not work !!
  -because of the few number of particles reaching the RoI,
  - filling the importance map to begin applying VRT takes a large CPU time (1000 particles reaching the RoI)
  - actualization of the importance map is extremely slow and uncertainties remain huge

MOSFET used as a in-vivo dosimeter



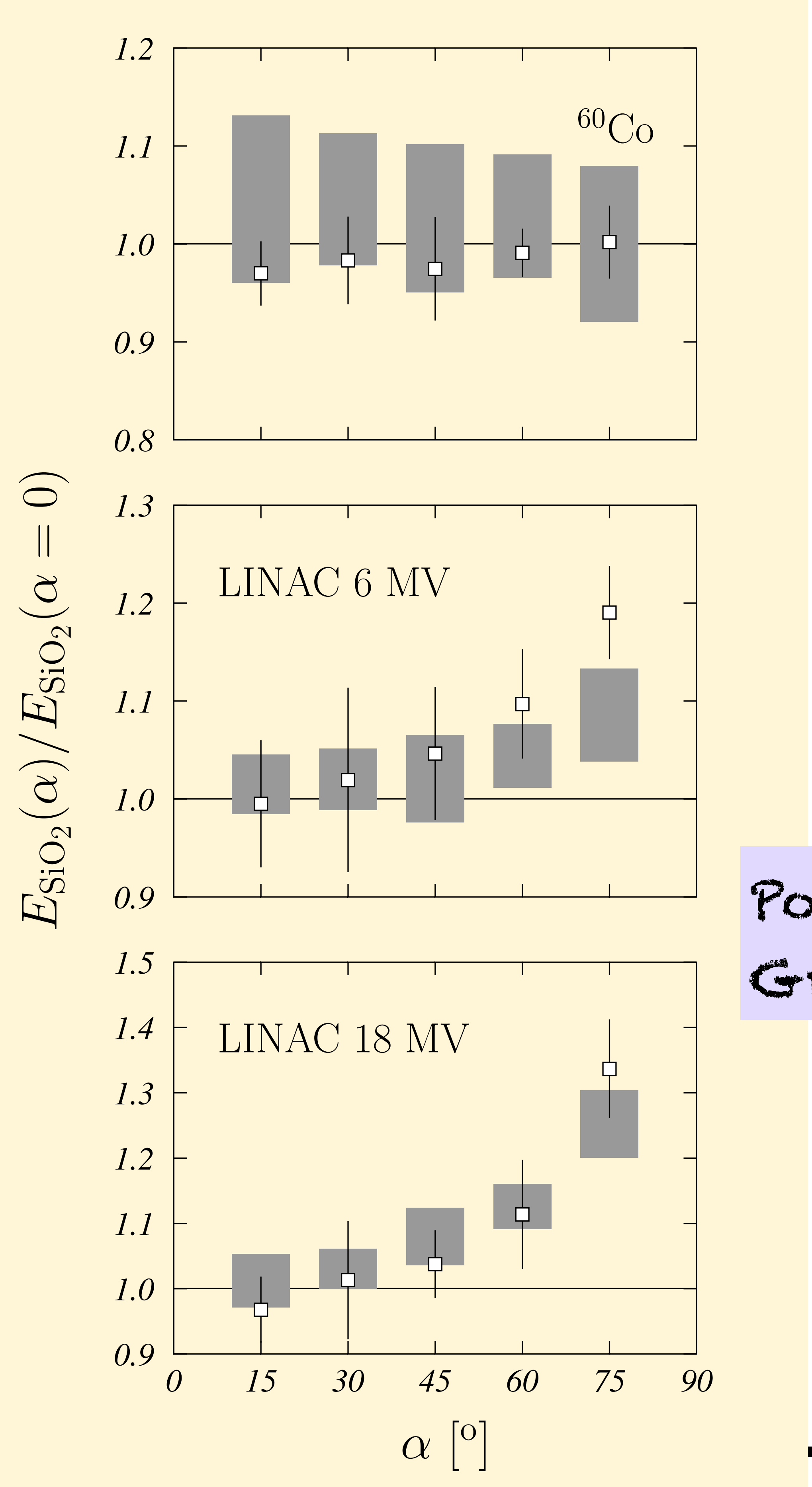
-address the possible angular dependence of the MOSFET response to irradiation

- If RoI is restricted to SiO_2
  - the algorithm does not work !!
  - because of the few number of particles reaching the RoI,
    - filling the importance map to begin applying VRT takes a large CPU time (1000 particles reaching the RoI)
    - actualization of the importance map is extremely slow and uncertainties remain huge

- optimal RoI:
  - whole Si die + SiO_2 + air
  - a gain factor ~70 to fill the importance map and "human" CPU times to reach reasonable uncertainties

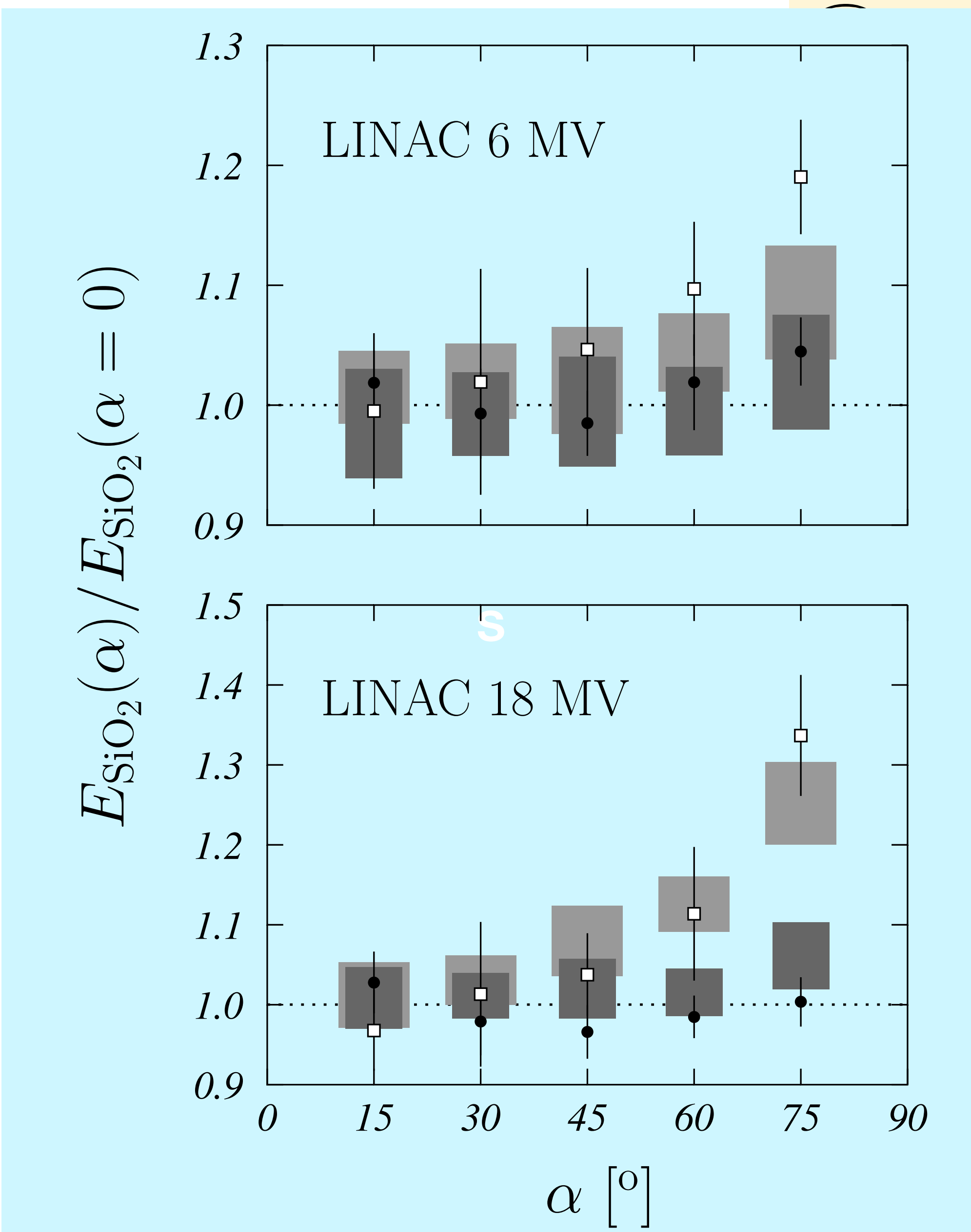
Results

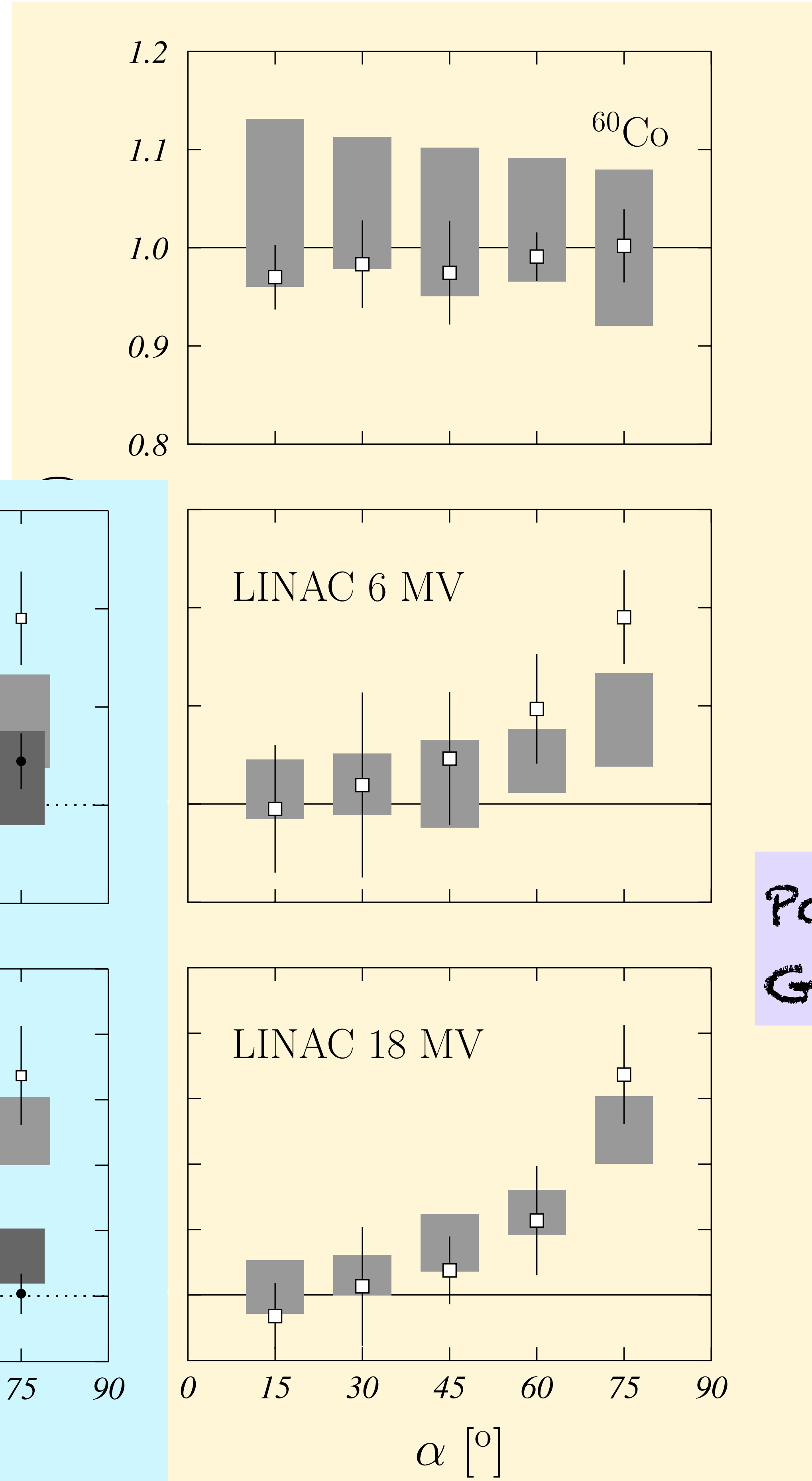
Results



Points: experiment
Gray bands: Monte Carlo

Results

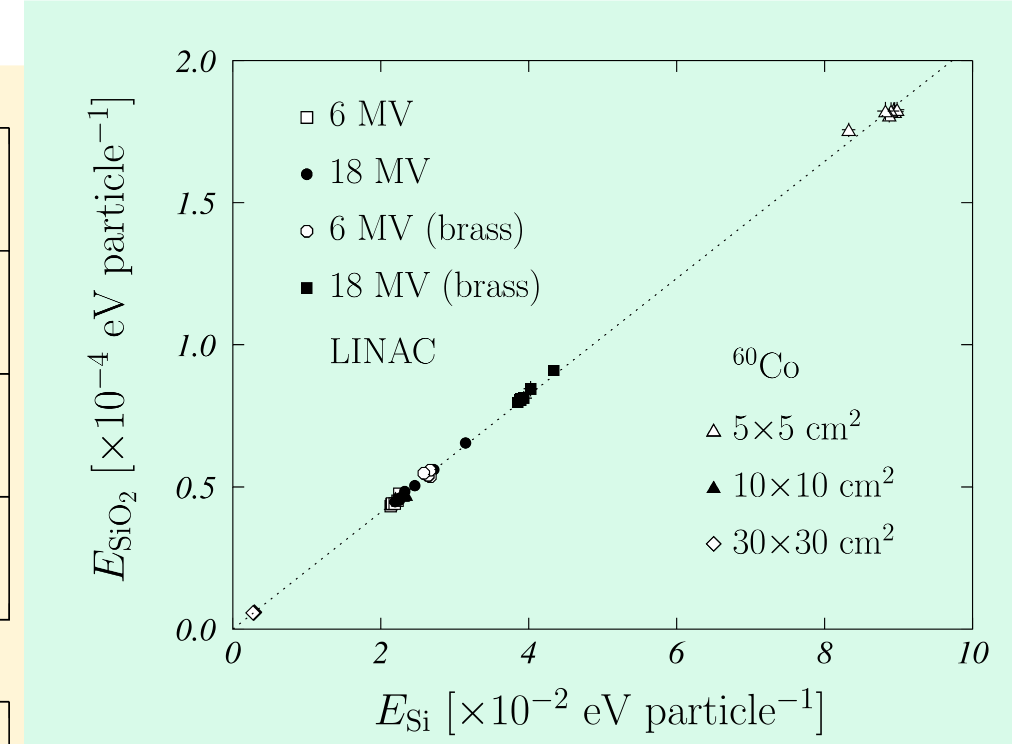


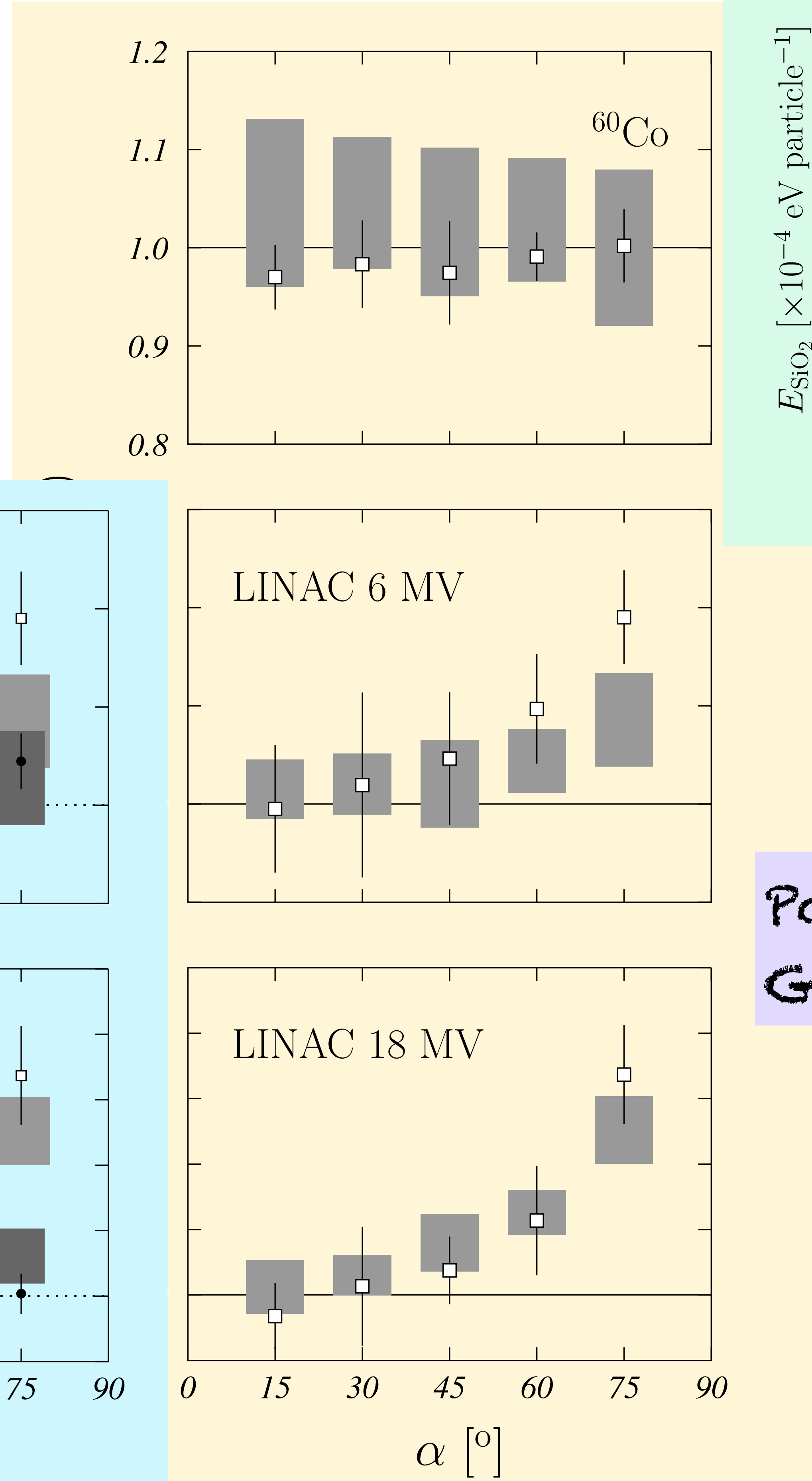


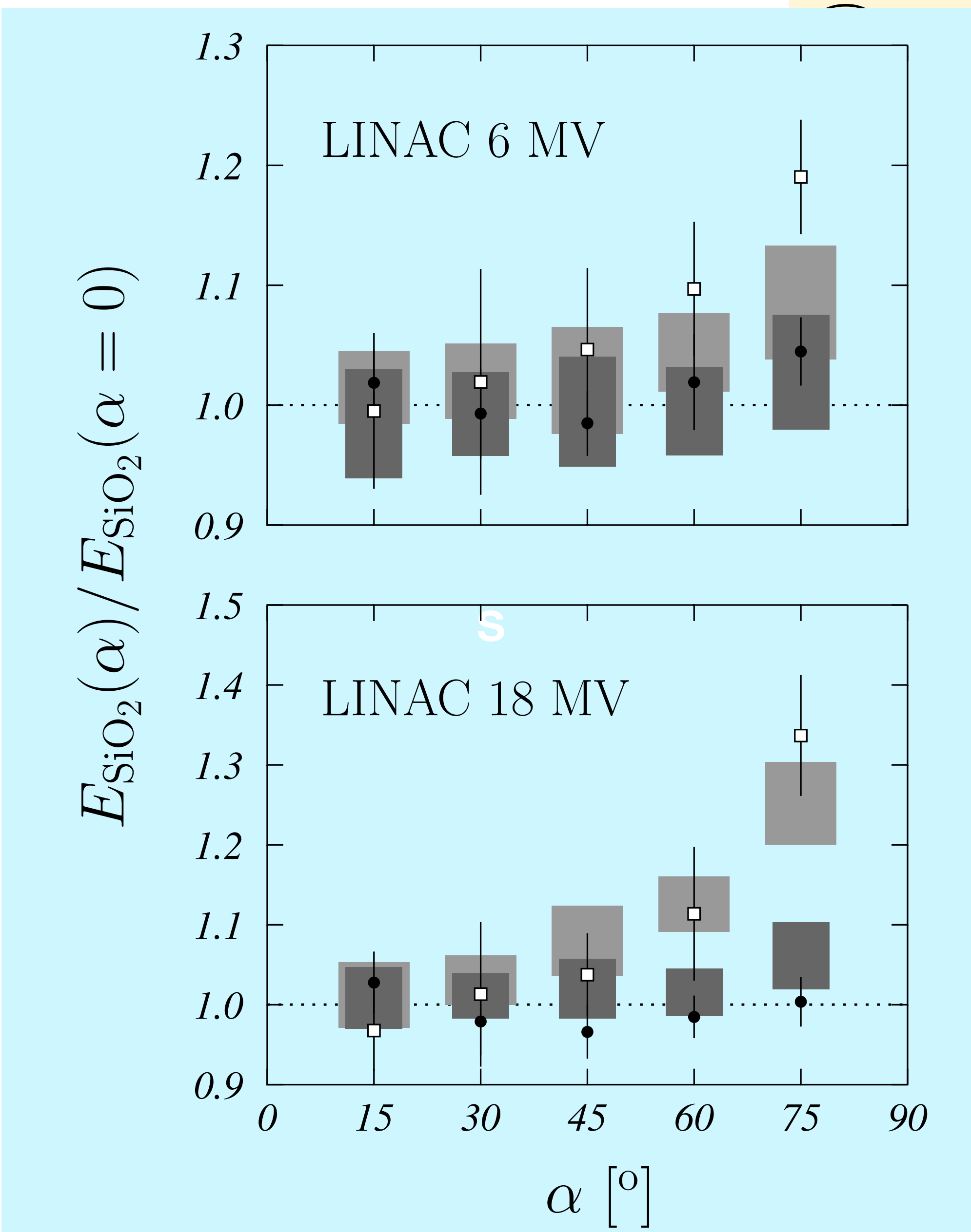
Points: experiment

Gray bands: Monte Carlo

Results







Points: experiment

Gray bands: Monte Carlo

•Conclusions

Conclusions

- VRTs allow simulating problems with very low statistics

- Conclusions

- VRTs allow simulating problems with very low statistics

- The ant colony algorithm developed allows the efficient implementation of VRTs by using the information scored on importance maps and with a minimum intervention of the user ... but details are relevant

Conclusions

- VRTs allow simulating problems with very low statistics

- The ant colony algorithm developed allows the efficient implementation of VRTs by using the information scored on importance maps and with a minimum intervention of the user ... but details are relevant

- Applications in other problems:
  - clinical linacs
  - radiosurgery photon beams (very small fields)
  - specific absorbed fractions (nuclear medicine)
  - correction factors of small ionization chambers

M. Anguiano Univ. Granada (Spain)
G. Díaz Londoño Inst. Tech. Metropolitan Medellín (Colombia)
F. Salvat Univ. Barcelona (Spain)
L. Brualla Universitatsklinikum Essen (Germany)
A. Palma, M.Á. Carvajal Univ. Granada (Spain)
D. Guirado Hosp. Univ. Granada (Spain)
F. Erazo SOLCA Cuenca (Ecuador)
M. Vilches IMOMA Oviedo (Spain)
P. Galán Hosp. Univ. Málaga (Spain)

Variance reduction techniques in Monte Carlo simulations: ants at work!

S. García-Pareja, A. M. Lallena

Thanks !!

Some applications

Photon beams for radio surgery

Some applications

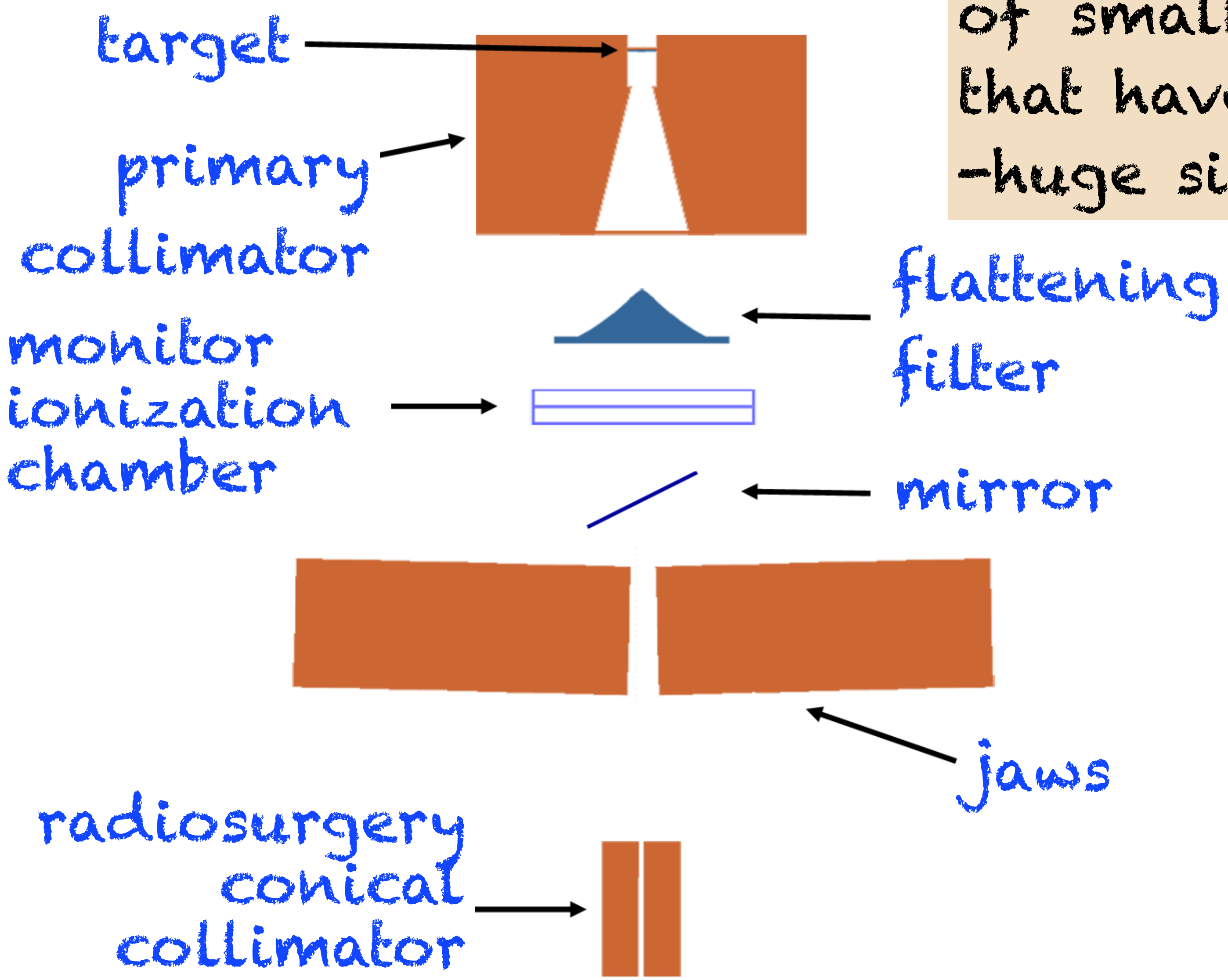
Photon beams for radio surgery

-very narrow beams used for treatment of small lesions nearby healthy tissues that have to be preserved
-huge simulation CPU times!!

Some applications

Photon beams for radio surgery

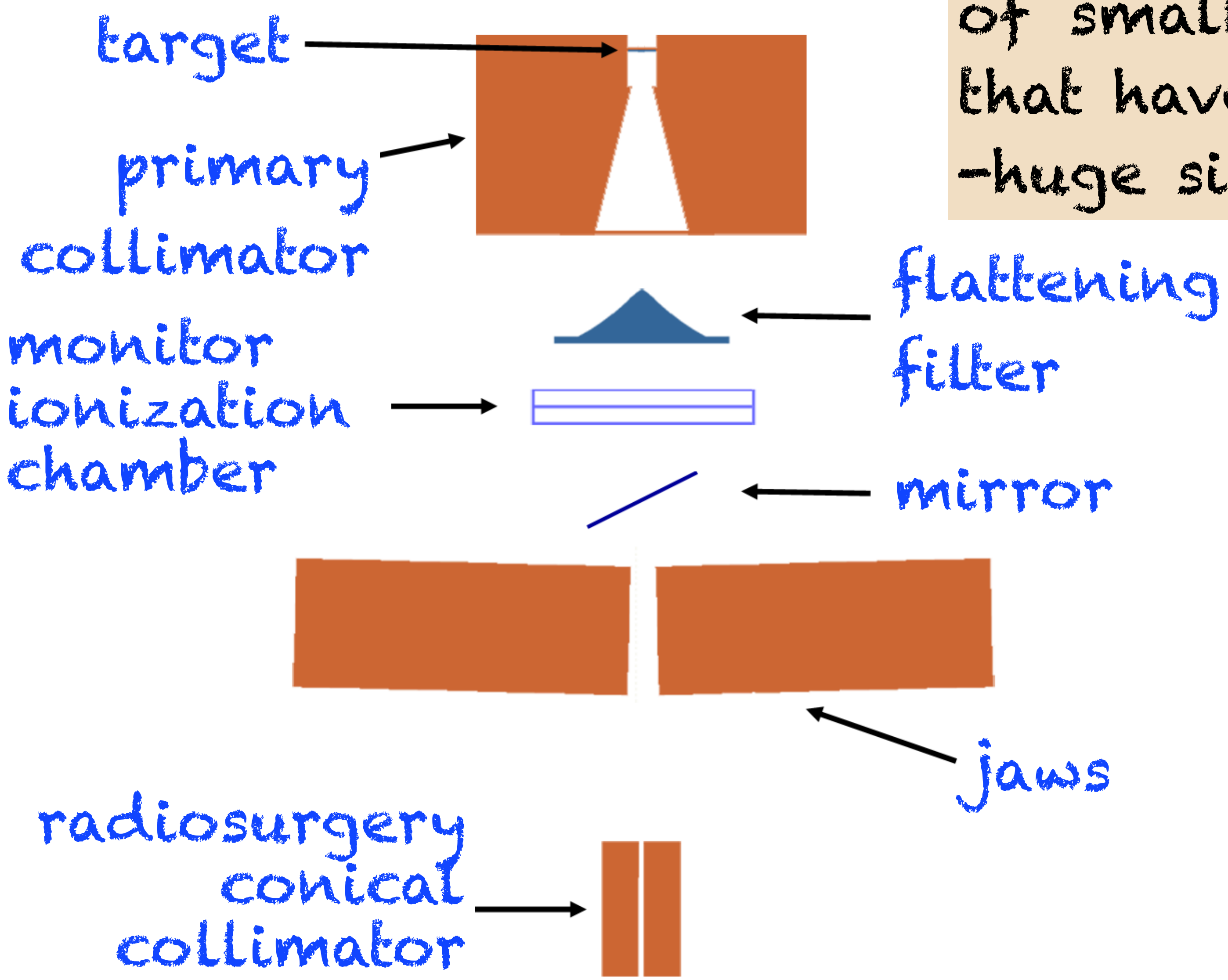
-very narrow beams used for treatment of small lesions nearby healthy tissues that have to be preserved
-huge simulation CPU times!!



Some applications

Photon beams for radio surgery

-very narrow beams used for treatment of small lesions nearby healthy tissues that have to be preserved
-huge simulation CPU times!!



-VRT applied to both electrons and photons

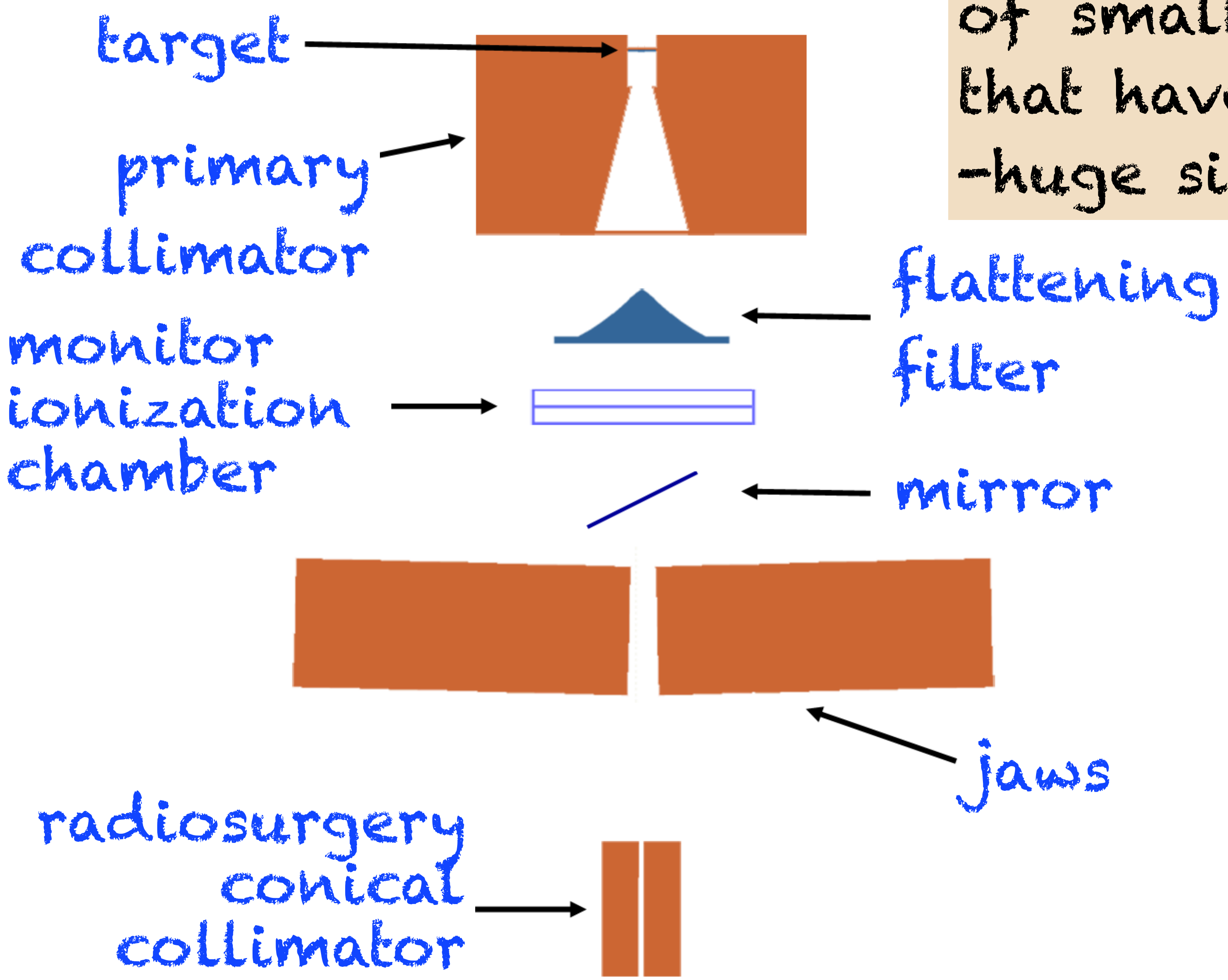
$I_1 = I(x, y, z, E, M)$ for electrons.

$I_2 = I(x, y, z, E, M, \theta, \phi)$ for photons.

Some applications

Photon beams for radio surgery

-very narrow beams used for treatment of small lesions nearby healthy tissues that have to be preserved
-huge simulation CPU times!!



-VRT applied to both electrons and photons

$I_1 = I(x,y,z,E,M)$ for electrons.

$I_2 = I(x,y,z,E,M,\theta,\phi)$ for photons.

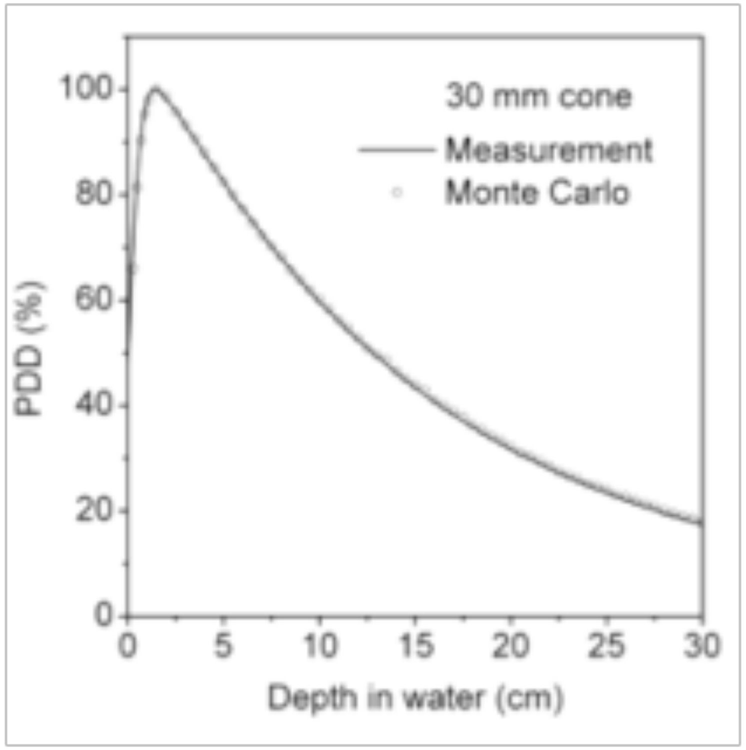
- directional Bremsstrahlung splitting is needed
  -applied throughout the whole geometry

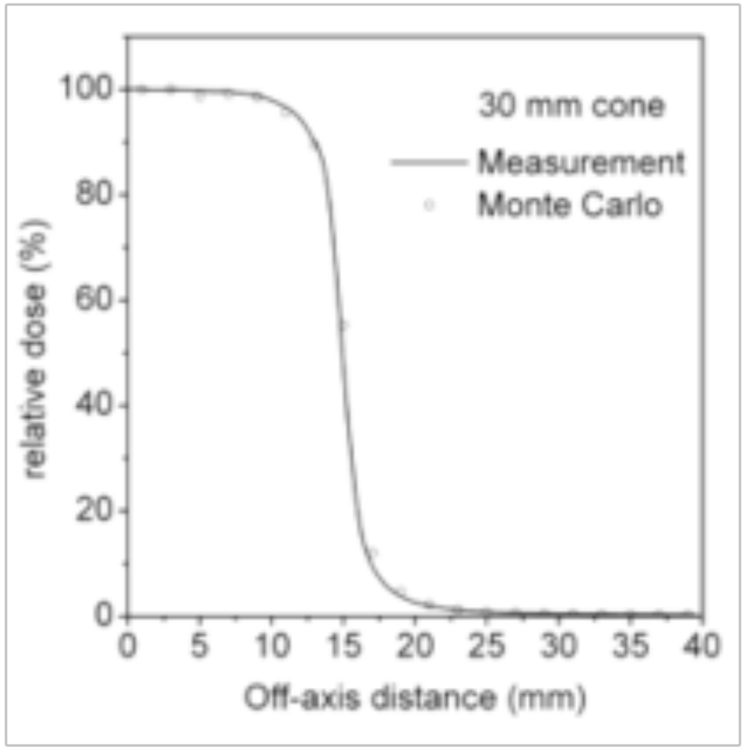
Some applications

Photon beams for radio surgery

Some applications

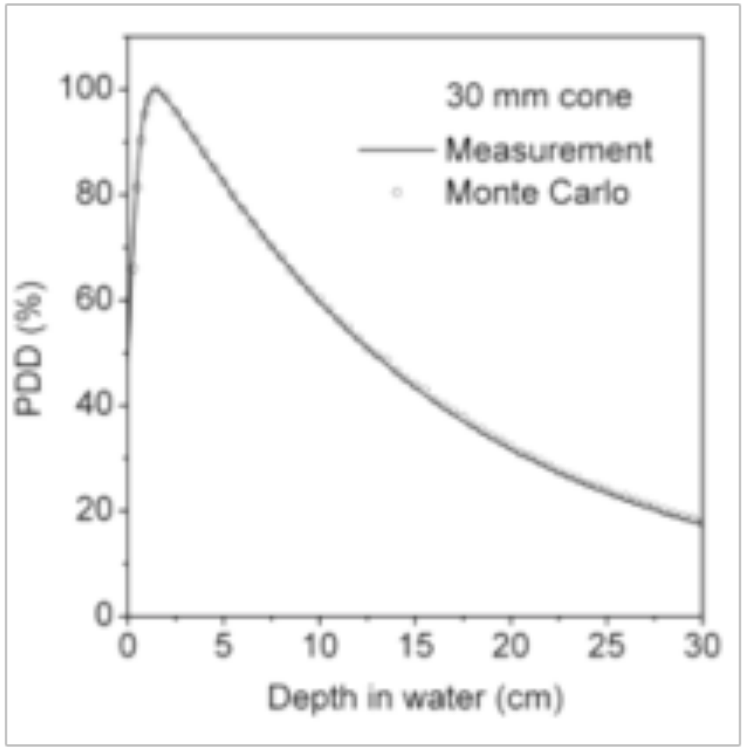
Photon beams for radio surgery

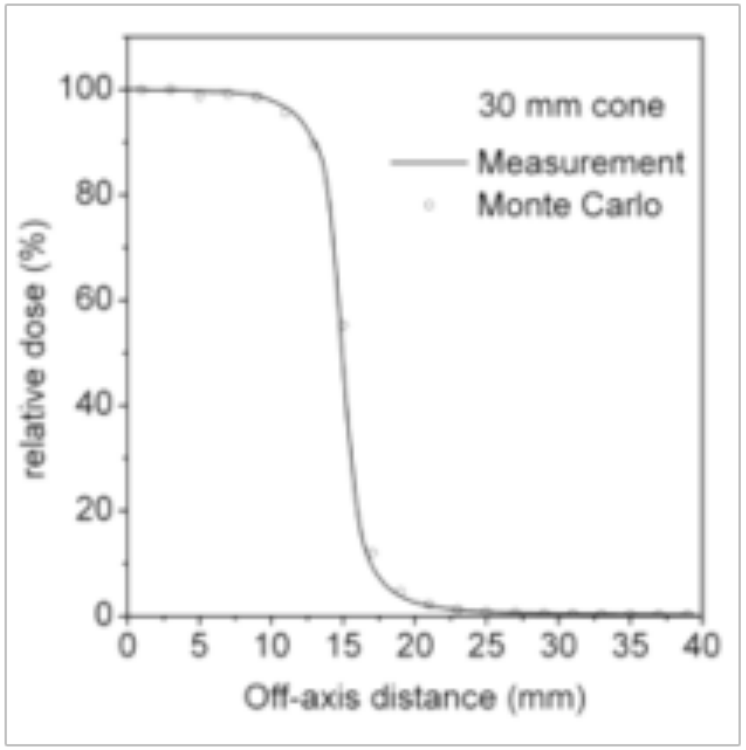




Some applications

Photon beams for radio surgery

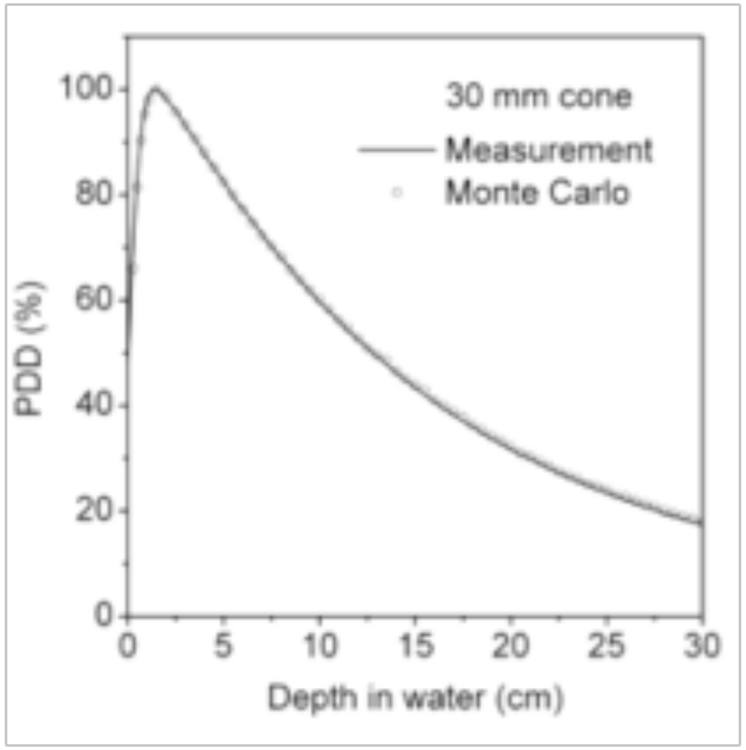


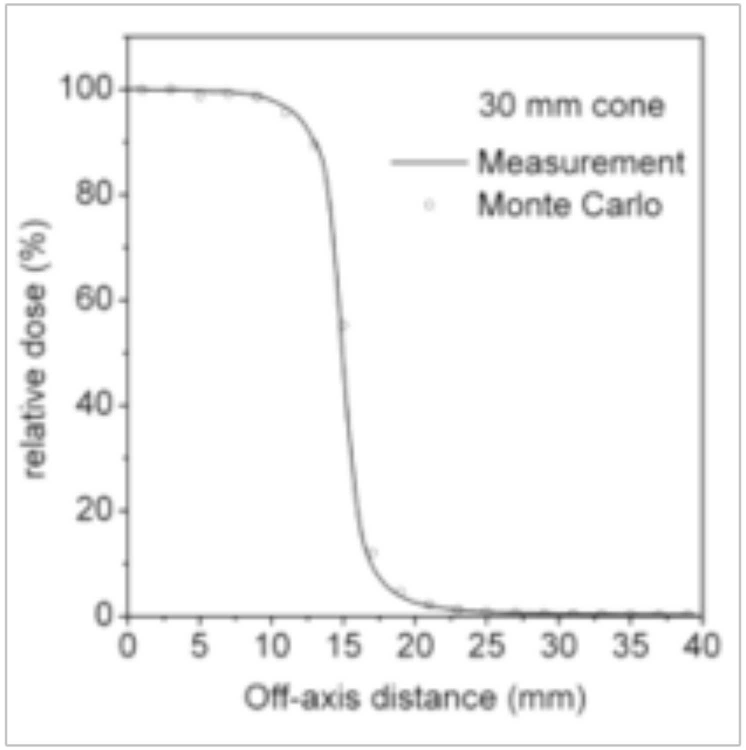


CPU times:
9 h (10 mm) to 0.9 h (30 mm)

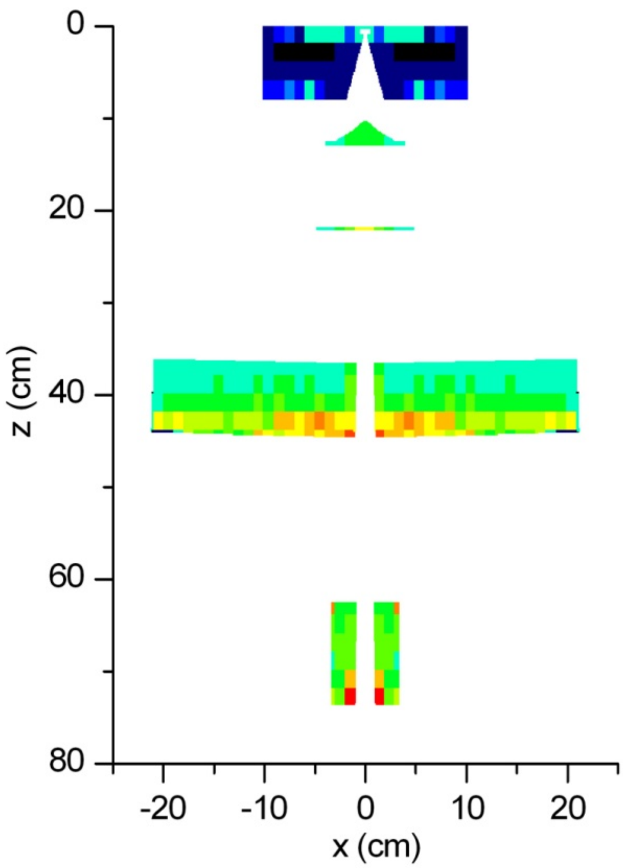
Some applications

Photon beams for radio surgery

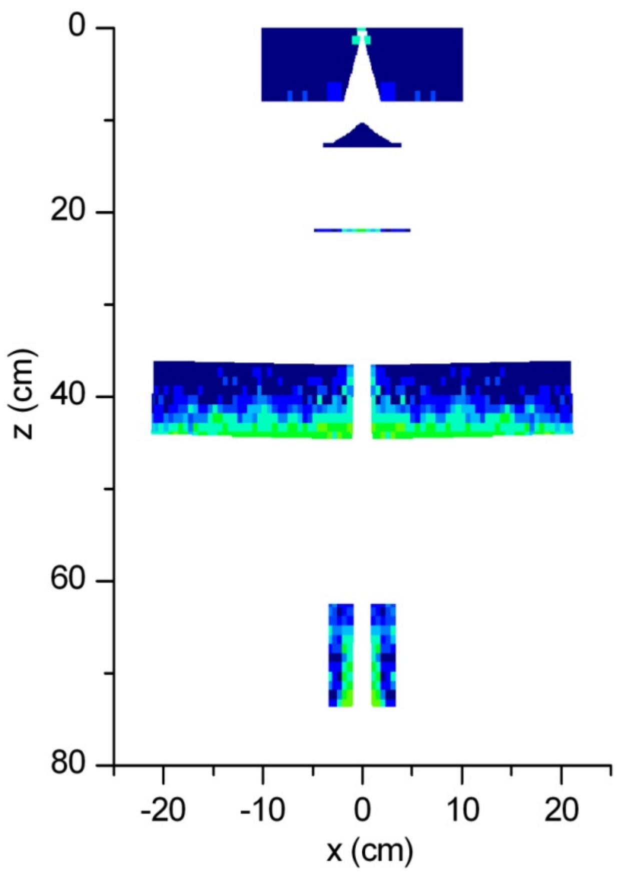




high energy photons



high energy electrons



CPU times:
9 h (10 mm) to 0.9 h (30 mm)

Some applications

Some applications

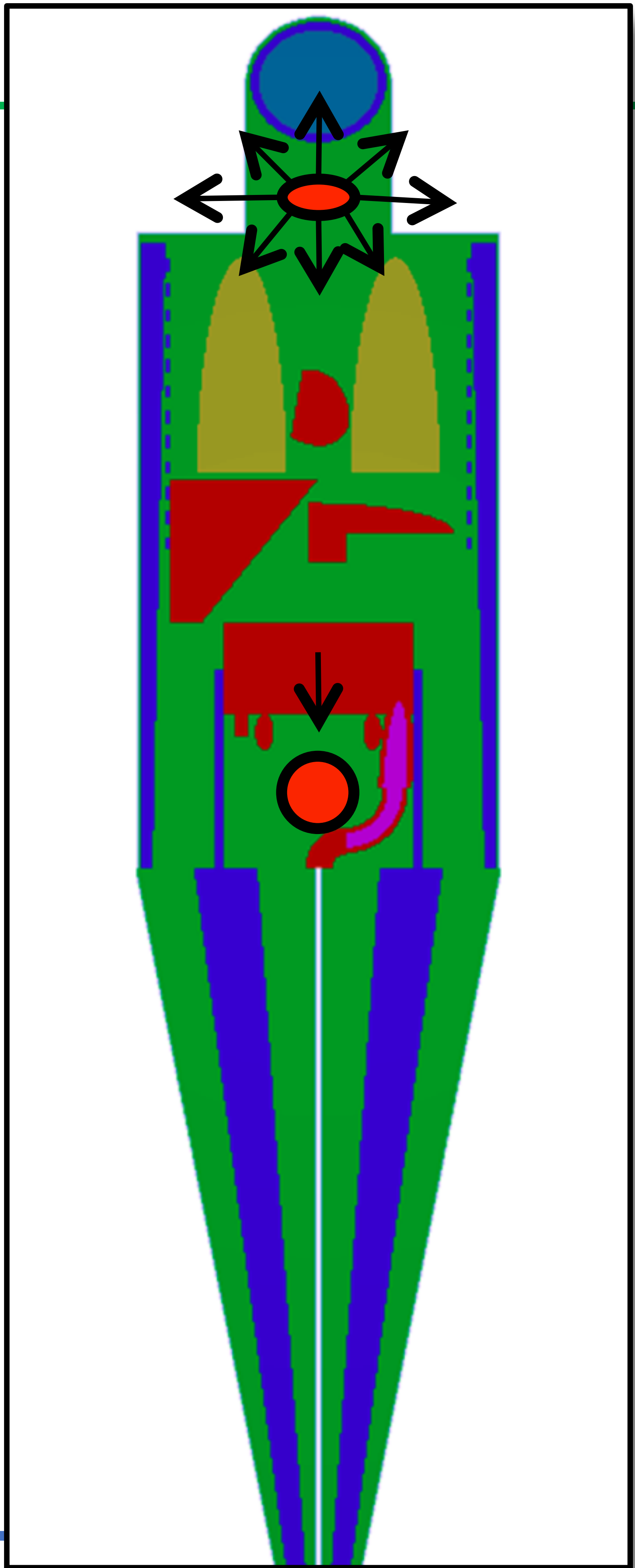
Specific absorbed fractions

Some applications

Specific absorbed fractions

-inform about organ/tissue irradiation due to diagnostic or therapy of other organ

-interest in Nuclear Medicine

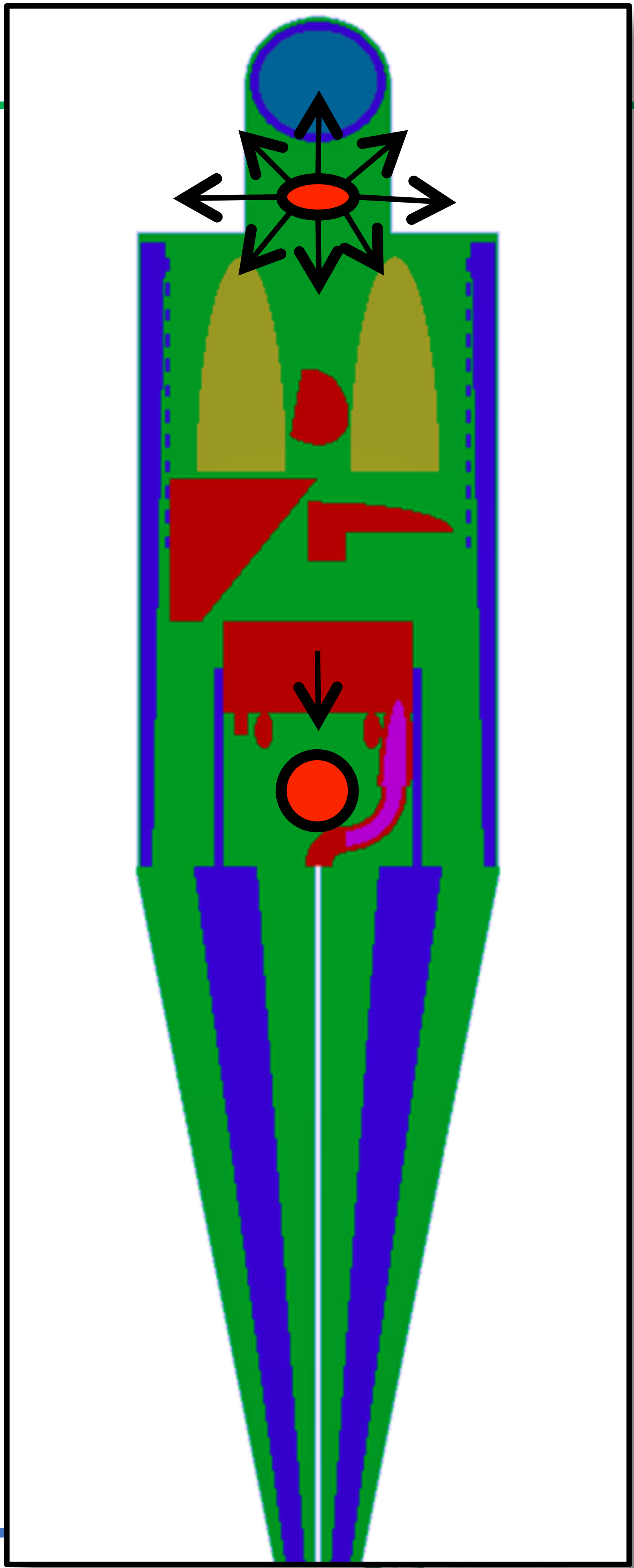


Some applications

Specific absorbed fractions

-inform about organ/tissue irradiation due to diagnostic or therapy of other organ
-interest in Nuclear Medicine

-problem: very low statistics because of organ volume and/or distance to source



Some applications

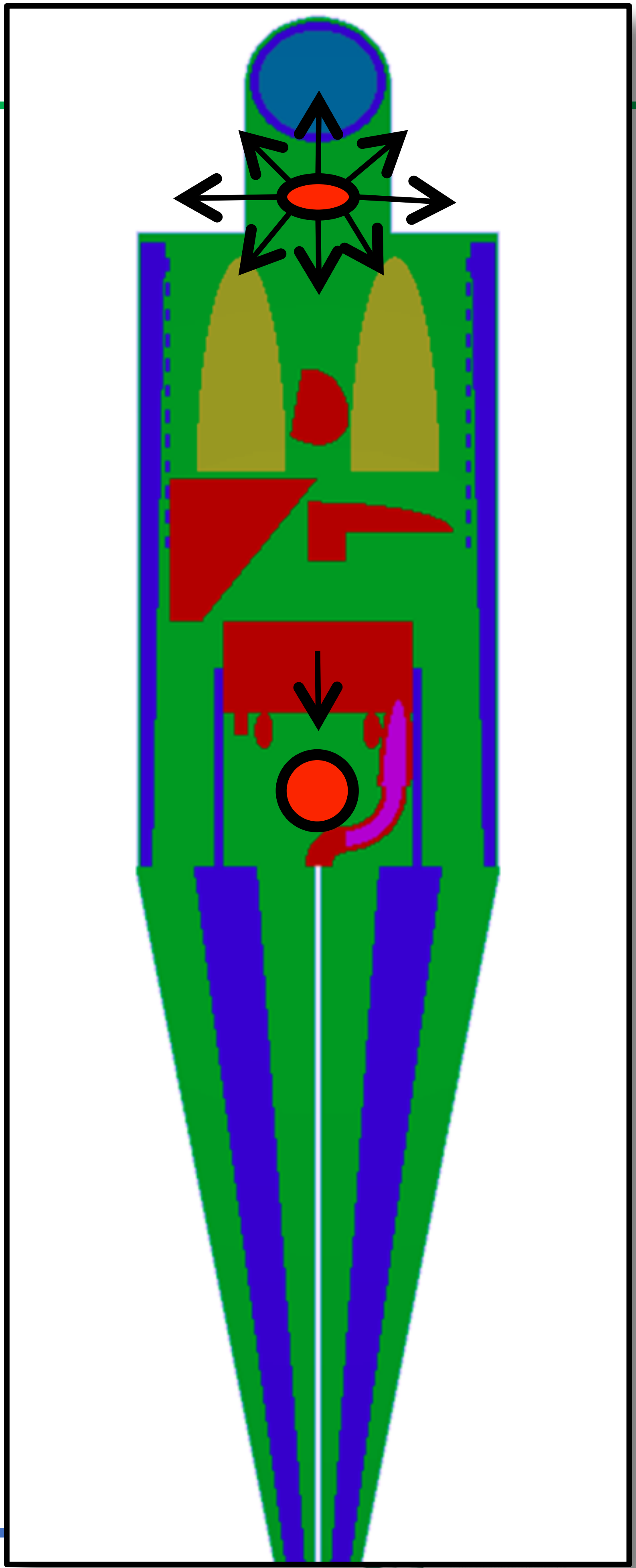
Specific absorbed fractions

-inform about organ/tissue irradiation due to diagnostic or therapy of other organ

-interest in Nuclear Medicine

-problem: very low statistics because of organ volume and/or distance to source

-efficiency increase by a factor 10!!



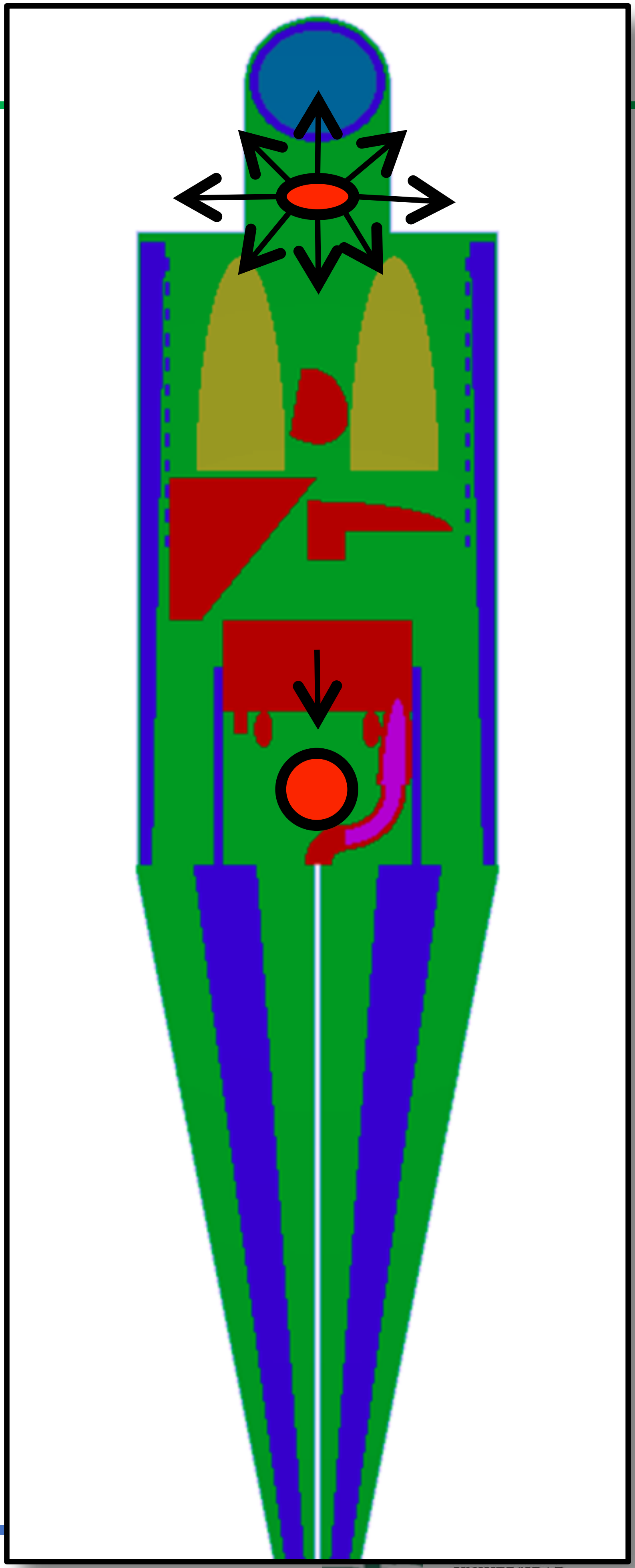
Some applications

Specific absorbed fractions

-inform about organ/tissue irradiation due to diagnostic or therapy of other organ
-interest in Nuclear Medicine

-problem: very low statistics because of organ volume and/or distance to source

-efficiency increase by a factor 10!!



Correction factors of micro-chambers

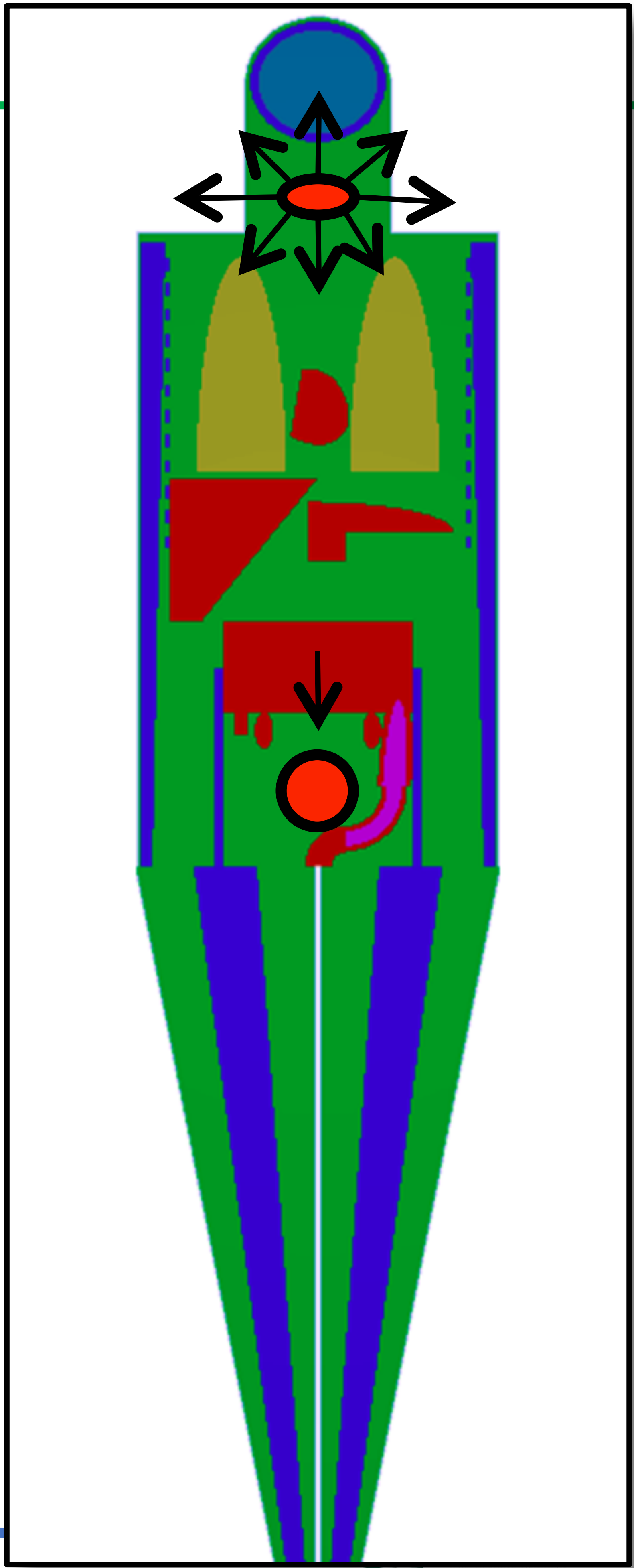
Some applications

Specific absorbed fractions

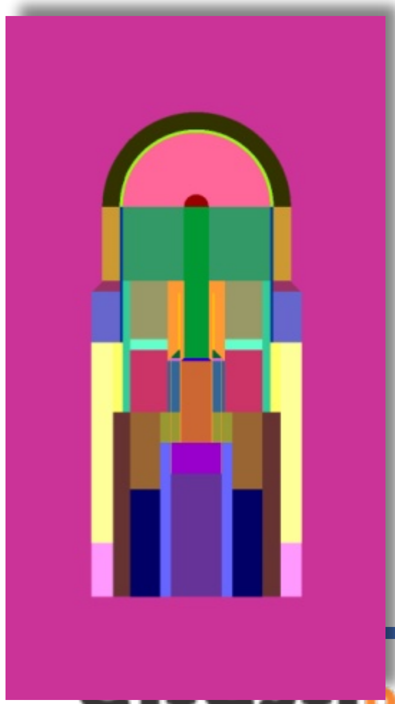
-inform about organ/tissue irradiation due to diagnostic or therapy of other organ
-interest in Nuclear Medicine

-problem: very low statistics because of organ volume and/or distance to source

-efficiency increase by a factor 10!!



Correction factors of micro-chambers



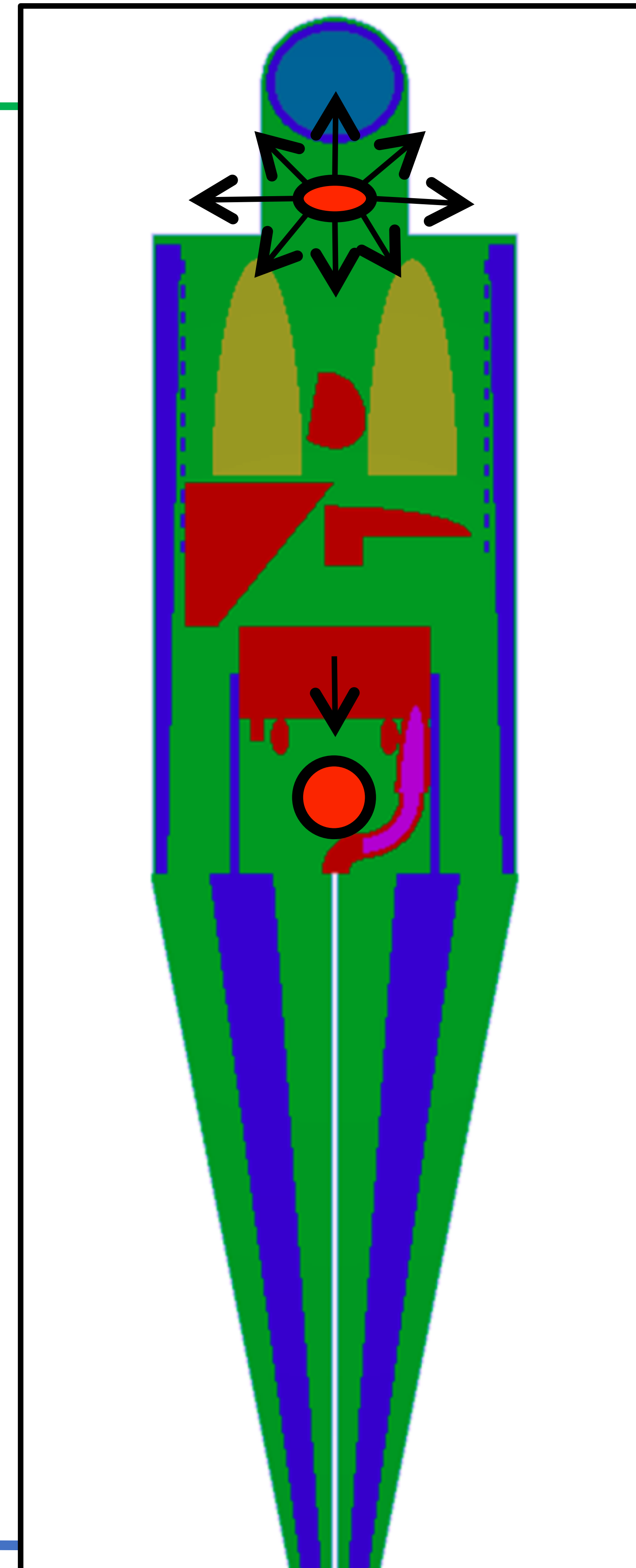
Some applications

Specific absorbed fractions

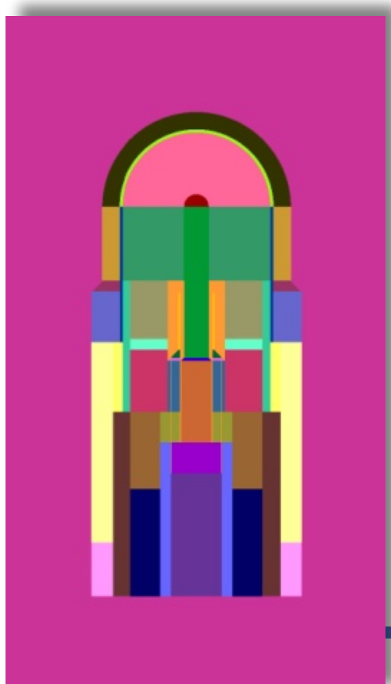
-inform about organ/tissue irradiation due to diagnostic or therapy of other organ
-interest in Nuclear Medicine

-problem: very low statistics because of organ volume and/or distance to source

-efficiency increase by a factor 10!!



Correction factors of micro-chambers



-efficiency increases by a factor 100!!

Conclusions

- An optimization algorithm based on 'ant colony behavior' has been developed
- It allows the efficient implementation of variance reduction techniques
- It uses the information scored on importance maps
- Minimum intervention by the user is required … but details are relevant