

Document downloaded from:

<http://hdl.handle.net/10251/38518>

This paper must be cited as:

Moltó Caracena, T.; Gonçalves, JGM.; Peerani, P.; Vendrell Vidal, E. (2013). A variable point kernel dosimetry method for virtual reality simulation applications in nuclear safeguards and security. *IEEE Transactions on Nuclear Science*. 60(5):3862-3871. doi:10.1109/TNS.2013.2279411.



The final publication is available at

<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6616662>

Copyright Institute of Electrical and Electronics Engineers (IEEE)

A Variable Point Kernel Dosimetry Method for Virtual Reality Simulation Applications in Nuclear Safeguards and Security

Teófilo Moltó Caracena, João G.M. Gonçalves, Paolo Peerani, Eduardo Vendrell Vidal

Abstract— This paper presents an algorithm to calculate gamma dose rates for Virtual Reality (VR) applications, which dynamically adapts the method to cope with both accuracy and time requirements. Given the real-time constraints imposed by VR applications, more accurate, but computationally intensive stochastic algorithms (e.g. Monte Carlo) cannot be used. On the opposite end, a Point Kernel (PK) method can be effective in some cases with as little as one point (Mono PK) to define a source, in contrast with the millions of points that Monte Carlo computes.

Simple Mono PK codes may lack the desired accuracy in some circumstances, requiring a more detailed source representation. In this work a novel method is presented which automatically estimates the appropriate level of detail for the source volumetric representation in order to reduce accuracy error. It then generates a non-regular mesh model and subsequently computes the dose rate via a PK method, performing this 3-step process in real time.

Index Terms—Virtual reality, computer simulation, gamma dosimetry, Point Kernel.

I. INTRODUCTION

Virtual Reality (VR) has been extensively used for training simulators in many areas of research and industry. Recent developments in the fields of Nuclear Security and Safeguards [1]-[3] have shown the potential of this technology for training applications, providing advantages in cost reduction and safety.

The success of the VR application simulation lies greatly on its capacity to provide a real-time immersive effect to the user. In practical terms this translates into two time-related requirements.

--First, the simulated instrument must calculate the dose rate in approximately the same amount of time than the real instrument does. This amount is not fixed because it depends on the type of handheld detector. Moreover, for a given detector, reading time might change depending on the energy

range being measured. This time typically ranges between 1 and 5 seconds.

--Second, the refresh rate of the visual rendering has to be as usual for VR applications about 30 frames per second (fps) in order to provide a fluid scenario movement.

Furthermore, considering this is a dosimetry simulation, another requirement is set aiming at providing an adequate accuracy of measurement:

--Third, the accuracy error in dose rate computation This value should be in line with that of the real instruments simulated. This value is typically, around 20%, which agrees with target values expected by IAEA [4] for this kind of non-destructive analysis.

The first requirement limits the algorithms to those who can be run in real time, therefore prohibiting computationally heavy codes like Monte Carlo, in spite of being optimal from the accuracy point of view, as other authors have agreed [5]-[7].

One of the already explored solutions to this problem is the use of a pre-calculated map of dose rates [8]. This method yields a very quick dose calculation based on interpolating the doses from the nearest points in the map. The limitation with this method is that interactivity with the scenario is limited, as changes in source or shielding elements will not be reflected on the dose rate. In this case, an off-line re-calculation is required. This limitation is not compatible with the smooth interactivity requirement one expects from a VR based training application.

In addition to the stochastic methods like Monte Carlo, deterministic methods can also be used in order to compute radiation transport. Among these, discrete ordinate methods (S_N) can provide a solution for simple geometries [9], nevertheless they are time consuming [10] therefore not being suitable for a real time application.

Alternatively, the deterministic PK methods can be used [11,12,13]. They are well known in the scientific community, and in particular in the nuclear physics field. They have been developed to facilitate scientists and engineers calculus of gamma dose rates since the 1980's. PK methods can cope with the requirements for this task, but on the downside they do not match Monte Carlo in terms of accuracy. Nevertheless given the large error margin conceded by the accuracy requirement, PK is suitable in some realistic cases considered in this paper.

Manuscript received August 9, 2012. This work was fully supported by the European Commission's Joint Research Centre PhD grant program.

T. Moltó Caracena, J.G.M. Gonçalves and P. Peerani are with the European Commission's Joint Research Centre, Institute for Transuranium Elements, Ispra, VA 21027 Italy (teofilo.molto-caracena@jrc.ec.europa.eu).

E. Vendrell Vidal is with the Universidad Politécnica de Valencia, Valencia, 46022 Spain.

We differentiate between the more simple and fast mono PK codes which only need one point to represent the source, like Nucleonica [11], and the Multi PK which use a model source composed by a set of points called mesh. Multi PK simply multiplies the number of operations by the number of points in the mesh. Therefore it is obviously slower (by a factor proportional to the number of points in the mesh) but it gives a better accuracy in cases where Mono PK leads to a too large dose rate error, namely when the detector comes closer to the source.

Many multi-point codes use a regular mesh of points to represent the source, i.e. PUTZ [12] or CIDEK [13]. These codes are aimed at shielding computations. These applications have different requirements and execution time is not a priority. Regular meshes with large number of points are configured offline, requiring users with expert knowledge in nuclear physics.

These existing PK methods cannot be used in VR simulations for dosimeter applications for the following reasons:

- 1) Source model selection: It is not evident to a non-expert user what level of point mesh resolution is necessary. The setup of the problem cannot rely on user input. It should be as automatic as possible as agreed by [14].
- 2) Dynamic variability of the model: In a VR application the user moves freely around the scenario, including possible interactions. These changes might require a variation of the source model representation. A VR application requires all these model modifications to be done in real time.
- 3) Efficiency of the source model: Regular uniformly distributed point meshes guarantee accuracy at the expense of unnecessary calculations. This is the case for the ones generated by the existing software. With this method, point density increases computational cost exponentially, and hence challenging the real time constraint. A more efficient point distribution is required to improve accuracy at lower computational cost.

This paper presents a prototype for a VR dosimetry simulation application, designed to comply with the previously stated requirements by combining the use of the well-known PK method with novel automatic, efficient, dynamic, real-time source model representation methods.

Next section explains in detail the prototype, along with the application structure, data flow and implementation details of the main elements: The PK dose rate computation function, the automatic resolution selector, the source model generator, and the VR interface.

The application was programmed using a VR oriented software development environment 3DVIA Virtools [15].

The prototype has been tested in order to verify the performance of the method, in terms of both (a) accuracy and (b) time response.

II. APPLICATION IMPLEMENTATION

A. Structure of the Application

The VR application is divided into two main parts:

--First, the dosimetry module which implements the dose rate computation and other dosimeter related functions. This module is executed periodically. Input data comes from the current status of the relevant 3D objects in the scenario. Outputs of this function are, both the 2D display representing the real display of the dosimeter, and the 3D rendering image of the source model in the scenario.

--Second, this part implements the user interfacing functions, translating input via keyboard and mouse into movement in the virtual scenario. This function updates data of the scenario's 3D objects which in turn are input to the dosimetry function. The output is the rendered 3D image of the scenario with the new position and orientation.

The following data flow diagram illustrates this scheme.

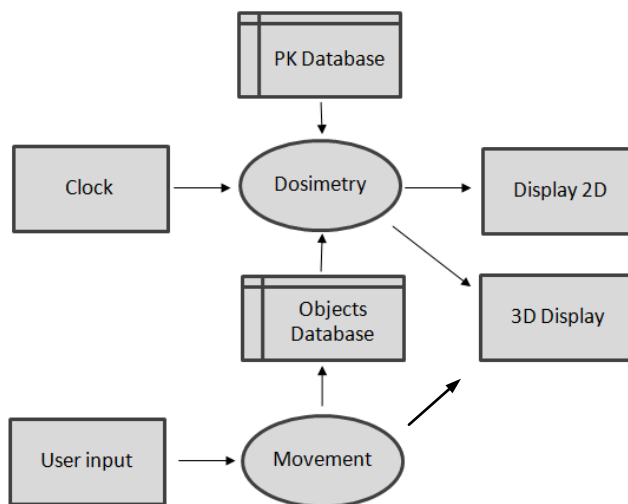


Fig. 1. DFD0: Data flow through main elements and input/output agents.

B. Dosimetry Module

The dosimetry module implements the three main functions of the application: Model Generator, Data Retrieval and PK Computation. These functions are executed sequentially in a continuous loop, together with a series of lesser priority functions.

Fig. 2 shows the data flow diagram of the *Dosimetry Module*.

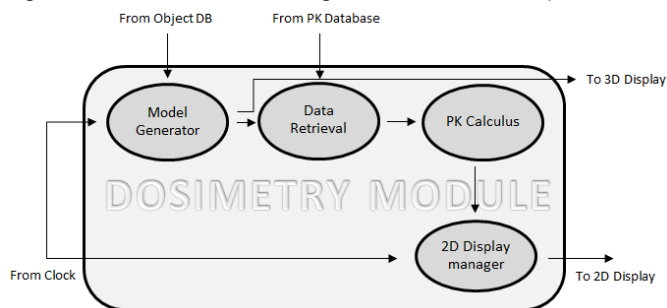


Fig. 2. DFD1 Dosimetry module

1) *Model Generator*: This module performs two tasks: first testing the current model of the source; second, generating the new model, if necessary.

It is not trivial to assess whether a model is sufficient in terms of dose rate computation error. There are several parameters that contribute to this error and require an increase in the point resolution of the mesh. Some parameters to consider are distance, size and orientation.

In this paper a single indicator, which can account for variations in all these three parameters, is used to test the validity of the source model. This indicator is the *solid angle*.

The *solid angle* is defined as a measure of how large the source volume appears to an observer looking from the point where the detector is.

The following figure gives a 2D example of cases that could require a source model change and how the solid angle increases when the case might require a higher point mesh resolution, therefore being a suitable parameter to evaluate the source model.

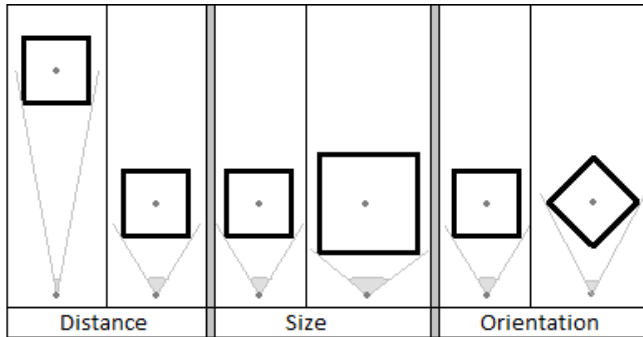


Fig. 3. Variations in Solid Angle detect changes in Distance, Size and Orientation, parameters used to evaluate source model mesh validity.

A threshold value for the solid angle is set in the application. This value was empirically set to 60°, resulting in a change of source model when that value is reached.

In this application, all sources are considered as parallelepipeds. This approach is valid in practical terms because most sources creating false alarms come in this shape (i.e. a pallet stacked with chemical fertilizer sacks or a truck container full of sand).

Given this representation, the *Solid Angle* function firstly computes the position of the eight points corresponding to the corners of the source, retrieving from the objects database the position of the barycenter, the orientation and dimensions of the source. In the next step it takes the position and aim direction of the detector. Then it computes the angle between the Detector-to-Source vector (normalized) and the Detector-to-Source-Corner vector (normalized). The algorithm calculates all eight angles and keeps the two largest ones on opposing sides of the plane defined by the detector-to-source vector and the detector position and adds them up. Fig. 4 illustrates this process simplified to 2D.

The application compares the calculated angle with the threshold value. If the calculated angle is lower, the current model of the source is kept as it is, otherwise the source will be passed on to the next function: *Cubic Division*, in charge of splitting the source.

The *Cubic Division* method generates eight parallelepipeds from each given source in the same way Octree methods work, that is, creating eight new source objects which inherit the original source properties, but with size halved in each dimension (X,Y,Z). The newly created objects are placed evenly spaced occupying the whole volume of the original source. For each new object the Activity (A) parameter is set to an eighth of the original source Activity so that the combination of all the eight new generated sources sum up to the same amount as the original source.

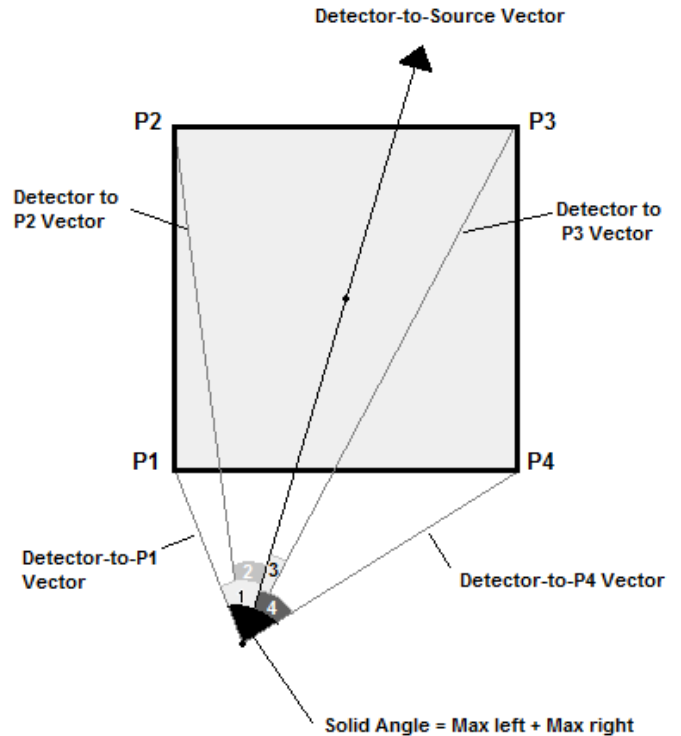


Fig. 4. 2D simplification of the Solid Angle calculus, addition of the largest angle on each side of the Detector-to-Source plane.

The new sources generated can be individually further divided in the same iteration. This generates an irregular division of the source. Unlike uniform point meshes, this method generates a source mesh, which is denser in the areas that are more significant for the computation. Fig. 5 shows an example of this division process simplified to 2D.

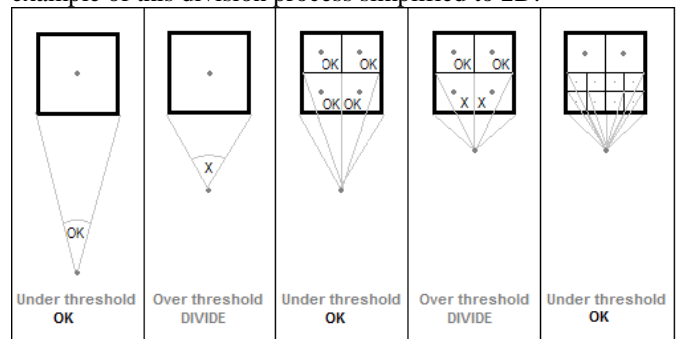


Fig. 5. The Cubic Division Method simplified to 2D. As the detector approaches the source, point resolution of the mesh is increased non-regularly.

Once every source point has been evaluated and all the sub-sources have a solid angle under the threshold, the model is completed for this iteration. The *Model Generator* phase is finished and the list of sources points updated.

2) *Data Retrieval*: This module has to provide all the necessary inputs to the PK Computation module.

The software development environment used, (3DVIA Virtools) implements a so-called *3D Object*. This object type includes useful attributes like position and orientation, and is easily rendered by the Virtools engine.

This object type definition is extended to include attributes necessary for the PK computation which the *Data Retrieval module* reads. These are:

- 1) Act: Activity of the source in Bequerels
- 2) Att: Link to Mass Attenuation Coefficient of the element tables in PK Database.
- 3) BU: Link to Build Up factor of the element tables in the PK database.
- 4) Density: Density of the element.
- 5) Obstacle: Faces of the geometrical model defined as obstacles for the Intersection calculus.
- 6) Spectrum: Link to Energy spectrum of the nuclide. Spectrum contains characteristic energy line and yield.

The *Data Retrieval* module starts by collecting the spectrum and activity data from the source object attributes. Subsequently it collects the positions of the detector and the source in order to calculate the distance between them. Next it runs the *Object Intersection* function.

The purpose of this function is to detect which obstacles might be in-between the source and detector (including the source itself) in order to apply the specific mass attenuation coefficient corresponding to that obstacle's composition.

It takes as an input the group of defined obstacles in the scenario, the position of the detector and source. Next it starts a loop to check if a ray traced between source and detector intersects each individual obstacle. If an obstacle is detected, the function takes its mass attenuation and Build up factor tables from the PK Database and the Density attribute of the obstacle. The thickness of the obstacle is calculated as the distance between the intersection points (entry and exit) of the ray traced through the obstacle.

In case no obstacle is found by this function, air is selected as the default media. The data gathered are sent forward to the PK Computation module.

3) *PK Computation*: This module performs the dose rate computation using the well-known existing Point Kernel method. To do so, it uses the received input collected by the *Data Retrieval* module plus the Absorption Coefficient. This is a parameter set by default to human tissue, but can be changed to air or silicon, if necessary.

The Point Kernel method implemented is based on two principles. First, computing the radiation intensity at the point where the detector is placed (I), based on the initial intensity at the source (I₀) and the distance between the source and the detector.

Second, estimating the attenuation due to possible obstacles in-between the source and the detector, including the source itself (self-absorption). This estimate is based on the following material's physical and chemical properties: thickness (d), density (ρ), and attenuation coefficient (μ). The following equation relates these factors.

$$I = \frac{I_0 e^{-\mu d}}{4\pi R^2} \quad (1)$$

This implementation of the Point Kernel algorithm takes into account an estimate of dose corresponding to the photons deviated by the obstacles by introducing a Buildup factor (B). This parameter is a function of the total attenuation coefficient, the thickness of the obstacle, and the energy of the gamma beam. Also, in order to compute the dose rate received by a certain material (detector, tissue, air), a mass energy absorption coefficient is included. Finally, the algorithm considers that there might be more than one spectral energy line of emission. Because of this, individual Energy/probability pairs have to be added.

Substituting in (1) the intensity at source factor (I₀) by the known parameters Source Activity (A) multiplied by the Energy of the gamma emission and its yield (Y) (probability of happening), (2) is obtained.

$$I = \frac{A \sum_{i=0}^n E_i \cdot Y_i \cdot B_i \cdot e^{-(\mu_i \cdot d) \text{ shield}} \cdot \left(\frac{\mu_i}{\rho}\right)_i^{\text{abs. coef.}}}{4\pi R^2} \quad (2)$$

This is the equation implemented in our application for the Mono PK case. Fig 6 illustrates a scheme of how the PK method is applied.

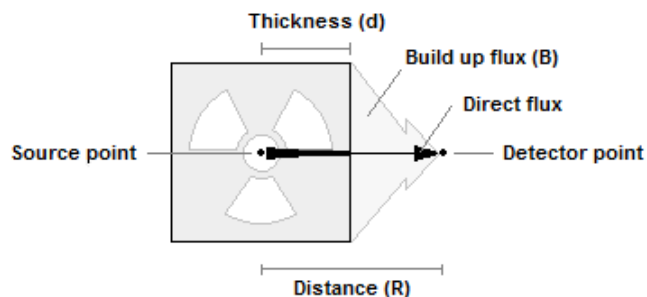


Fig. 6. Illustration of Point Kernel scheme. Illustrates the disposition of the elements that intervene in the calculus of (1)

When there is an increase in the point resolution of the source, a multi-Point Kernel method is used. This alternative creates an extra loop, so for each iteration, every point is treated like a new source, and the mono PK method is applied to each mesh point. Finally, the contribution of each mesh point is added up to obtain the total dose rate of the original source.

Starting from (2), the equation is modified by including an outer summation loop for each point (j) of the mesh, dividing the total activity by the number of points in the mesh, and substituting the distance to the center of the original source (R)

by each individual distance (R_i) from the detector to each point (j), resulting in (3).

$$I = \frac{\sum_{j=1}^{j=\text{points}} A_j \sum_{i=0}^n E_i \cdot Y_i \cdot B_i \cdot e^{-(\mu_i \cdot d) \text{ shield}} \cdot \left(\frac{H_i}{\rho}\right)_i \text{ abs. coef.}}{4\pi R_j^2} \quad (3)$$

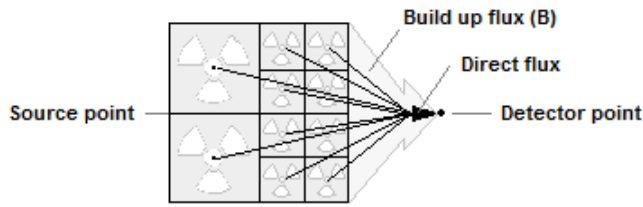


Fig. 7. Illustration of Multi Point kernel scheme simplified to 2D. A mesh of points represents the radiation source, in this case a 10 point mesh substitutes the single point of Fig 6. The Mono Point Kernel method is applied to each point and the individual doses added up.

Build-up factors (B), mass attenuation coefficients (μ_i) and mass absorption coefficients (μ_a) included in the PK Database are obtained by performing linear interpolation from values available from the well-known ANS data tables [16].

Energy value ranges covered in these tables are detailed in the Code Limitation section. If the requested values lie outside the range of energies covered by these tables, a linear extrapolation function is applied.

The *PK Computation* implements a binary search on these data tables to find out the values for the linear interpolation for the three parameters needed. Using these three parameters and the ones passed by the Data Retrieval module, the *PK Computation* function can finally apply the formula previously described to compute the dose rate. The result is the dose rate in mSv/h, which is made available to the interface related modules described in the following point.

4) *2D Display Manager*: This module includes all the functions in charge of simulating the display of the detector. The following figure shows the data flow of this sub-module, which contains five individual functions.

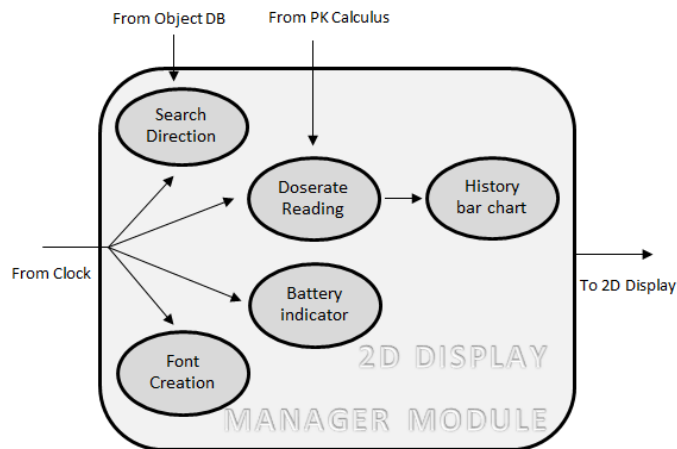


Fig. 8. DFD2 2D Display Manager Module

--First, the *Search Direction* function simulates a visual indicator of the direction in which the source lies with respect

to where the detector is being pointed at. It is one of the features of the real dosimeter being simulated. Two horizontal bars, grow or shrink accordingly, indicating the angle to the source as a percentage. Each bar measures from -90° (0%) to 90° (100%). When both bars are at 50% it means that the source lies straight ahead.

If the angle is larger than 90° the bars are set to zero indicating the user that the source is behind. This function takes the information about the position and direction of the detector, and the position of the source from the internal object database. Then, it performs a simple trigonometric computation and converts the resulting angle into a size for the horizontal bars.

--Second, *Doserate Reading* is a simple function, but the most significant. It takes as an input the dose calculated by the PK Computation function in the last iteration and displays it on the screen of the 2D dosimeter replica display. By default, it uses $\mu\text{Sv/h}$ as dose rate unit, but the user can switch to mRem as many real instruments do.

--Third, the *Waterfall Chart* function indicates the magnitude of the last few dose readings in a visual way (the vertical bars represent the magnitude of the measurement). This helps the user to locate the source, as the magnitude will be directly proportional to the distance for a given source. An increasing set of bars corresponds to an approach to the source, while decreasing bars means the user is pulling away. The function simply stores the last ten dose rate values, which are provided by the Dose Rate reading function. The values are converted to bars of proportional size. For each iteration, the bars are shifted leftwards, the oldest measurement is disregarded, and the newest is placed on the rightmost part of the display.

--Fourth, the *Battery Indicator* function is a simple function in charge of visualizing the amount of battery lifetime left. Battery lifetime is set according to the manufacturer’s specifications of the detector.

--Fifth, the *Font Creations* is an auxiliary function that creates and customizes the fonts used in the different sections of the 2D display (Search Direction, Doserate reading and Battery indicator).

C. Movement Module

The final module of the application covers the movement of the user within the scenario.

The main function waits for user input on keyboard and mouse. Mouse movement is translated into user’s viewpoint orientation change, allowing him/her to “look around” the scenario. Keyboard input translates into movement forward back, left, right and rotation clockwise and counter clockwise, as shown on the next figure.

This configuration replicates that of other 3D non-related software (i.e. videogames) so that untrained users might find it familiar to use.

An orbiting function is also included. This allows the user to circle around the source without changing the distance and noticing in real time the effect of being behind a shield or not.

The current prototype uses mouse and keyboard, but other VR interfaces like head mounted displays can be used in order to enhance the user's immersive experience if necessary.

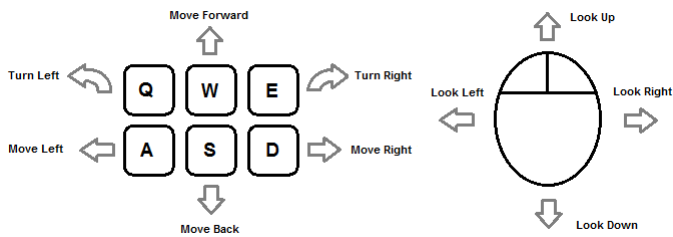


Fig. 9. User input controls, mouse for view change and keyboard for movement.

Finally a very simple hide function allows the user to hide the detector from the display to get a better view of the source. The rendering engine of Virtools converts these movements into the appropriate 3D view.

III. USER INTERFACE

A dosimetry training application for a non-expert audience requires a simple interface isolating the user from the complex physics processes associated with the task.

Existing dosimetry applications do not provide this required simplicity. Their use requires a great deal of expertise both in nuclear physics and in the application's configuration and set up.

A feature of the current VR based interface is the clear separation between the use of the instrument and its underlying working principle.

This makes this interface suited for training applications, namely when trainees are not supposed to have a deep understanding in radiation physics, as it is the case for customs officers, border guards or emergency first responders.

As in most VR simulators the user is provided a 3D first person view, enhanced with a 2D replica of the detector's display located on the top left corner of the user's screen. This feature is implemented in order to facilitate the reading of the instrument by the user. Fig. 10 shows this view.

Data on the 2D display is updated periodically, in order to replicate the typical measurement frequency of this kind of instruments. This display shows the info generated by the *Doserate function*, *Waterfall Chart*, *Direction* and *Battery Indicator* functions, as shown on Fig. 11.

The user can move around the virtual scenario by using an appropriate movement interface, while the application computes the dose rate in a manner completely hidden to the user. He/she does not need to understand the physical models underlying in the application, therefore being suitable for non-expert users.

The user may interact with some objects of the scenario, i.e. the two walls visible in Fig. 10 providing a shielding behavior. An object sliding function prevents the user from walking through objects and a keep on floor function keeps the camera point of view at a constant height.



Fig. 10. User view of the application. A cubic source in the center. The walls around can be used to appreciate the result of the shielding (in real time) in the dose rate reading if the user walks behind them.

The 3D rendering of the scenario is generated by the CK2_3D engine provided by the development software platform (3DVIA Virtools Version 5.0).

3D rendering is not a subject of this study, and given the fact it only takes constantly approximately 0.5ms per frame on the tested computer (barely 1% of the limit stated in the time requirements) we will disregard its effect for evaluation purposes.

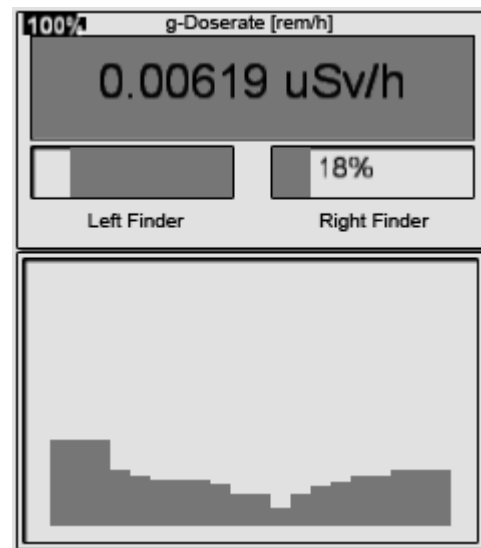


Fig. 11. 2D Display virtual replica, top left corner shows Battery Indicator, top center Dose rate reading, below are Direction indicators and at the bottom the Waterfall chart is placed.

IV. RESULTS

Two sets of tests have been performed to validate this VR based application. The first set tests the VR application against real measurements taken with a real gamma dosimeter in a typical nuclear security training exercise case. The second set

of tests compares the VR application against other software codes for a reference nuclide case.

A. Case 1: Fertilizer stack (⁴⁰K source) real measurements

The first case considers the measurement of the dose rate generated by a stack of Potassium Chloride (KCl) fertilizer sacks. This source is chosen as a test case because it is a typical cause of false alarms at radiation portals at international border crossings. Therefore this is one of the sources that custom officers need to learn to detect in training sessions.

In order to setup the test for the VR simulation, the activity of the source, the density and the geometry need to be defined.

The manufacturer guarantees that over 95% of the fertilizer is KCl, therefore we assume the whole mass of 750kg to be KCl. Taking the data mass numbers, isotopic abundance and activity of ⁴⁰K from [17] we define the activity of the source as 12.54 MBq.

To define the geometry of the source, the stack was measured and represented as a 110cm long, 110 wide and 60cm tall parallelepiped. From this measurement, the volume (110*110*60 = 726000 cm³) was inferred and, given the mass stated by the manufacturer, the density was calculated (1.033 g/cm³).

The PK library is updated with a new table for KCl mass absorption coefficient, where all values are estimated as an average of the existing K and Cl table values. The 1460keV energy line was considered with a yield of 10.72%

For the real measurements a handheld gamma ion chamber survey meter is used (manufacturer Fluke model Victoreen 451P). This apparatus detects gamma radiation above 25keV. It has a response time which varies between 1.8s and 5s depending on the operating range measured. The manufacturer states an accuracy of +/- 10% of the reading between 10% and 100% of the range.

Measurements were taken at different positions varying distance and orientation (front, above, diagonal) of the source, and considering the origin of coordinates in its center, as shown in the next figure.

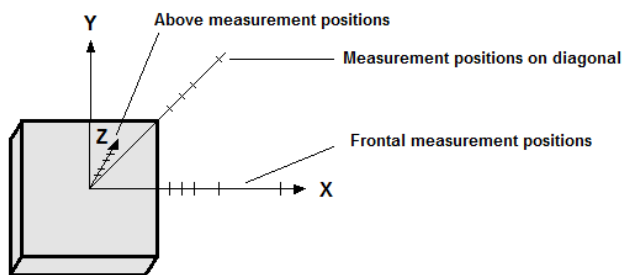


Fig. 12. Measurement positions for Test 1 KCl source.

In first place accuracy is tested. The following two tables show the dose rate results obtained from the measurements, those obtained by simulating, and the deviation expressed as a percentage.

Table I shows the results for a mono PK (the simplest already existing PK method) version of the VR application.

TABLE I
REAL DETECTOR VS. MONO PK SIMULATION

Distance [cm]	Victoreen 451P [μSv (Air)/h]	Mono PK [μSv (Air)/h]	Deviation
Frontal			
10	0.37	0.127	66%
20	0.23	0.095	59%
30	0.17	0.075	67%
50	0.09	0.049	46%
Above			
10	0.48	0.778	62%
30	0.25	0.346	38%
50	0.18	0.194	7%
Diagonal			
10	0.19	0.0293	85%
20	0.14	0.0236	79%
30	0.1	0.0194	89%
50	0.06	0.0138	77%

Table II shows the dose rate results using the new Variable non-regular source representation method developed in this work.

TABLE II
REAL DETECTOR VS. NEW PK SIMULATION

Distance [cm]	Victoreen 451P [μSv (Air)/h]	VR Application [μSv (Air)/h]	Deviation
Frontal			
10	0.37	0.367	1%
20	0.23	0.254	9%
30	0.17	0.185	9%
50	0.09	0.095	5%
Above			
10	0.48	0.484	1%
30	0.29	0.289	13%
50	0.18	0.184	2%
Diagonal			
10	0.19	0.166	13%
20	0.14	0.109	22%
30	0.1	0.085	15%
50	0.06	0.042	30%

The following graphs illustrate the deviation trends of the new (variable Multi PK) and old (Mono PK) with respect to the reals measurements in the three testing configurations.

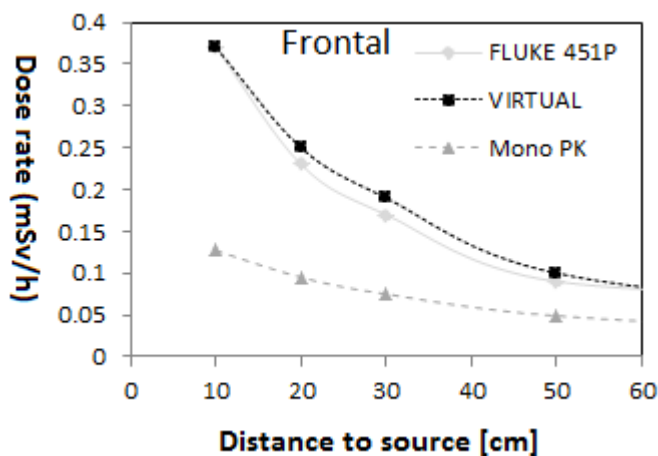


Fig. 13. Dose rate comparison, Real measurements vs. New (Variable) and Old (Mono PK) Simulation methods, frontal measurements.

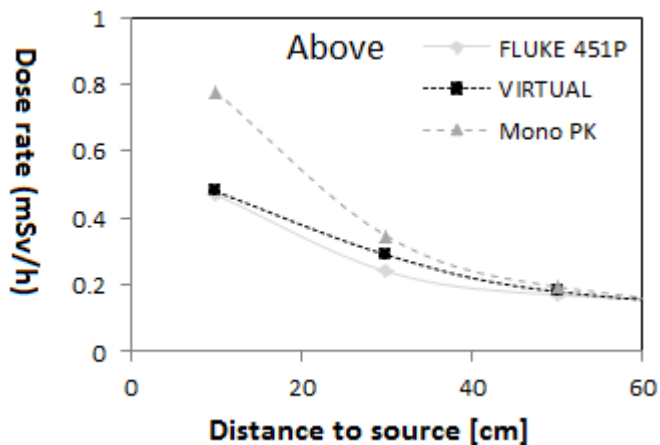


Fig. 14. Dose rate comparison, Real measurements vs. New (Variable) and Old (Mono PK) Simulation methods, above measurements.

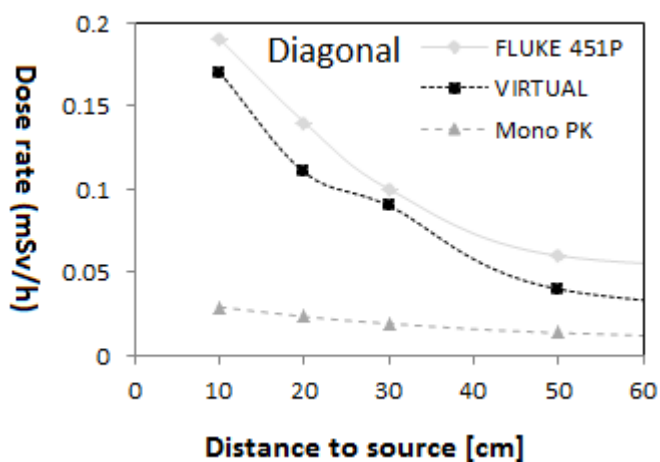


Fig. 15. Dose rate comparison, Real measurements vs. New (Variable) and Old (Mono PK) Simulation methods, diagonal measurements.

Using this method at the closest simulating distance, the application has generated a total of 120 sub-sources in four different levels of resolution as Fig 16 illustrates.

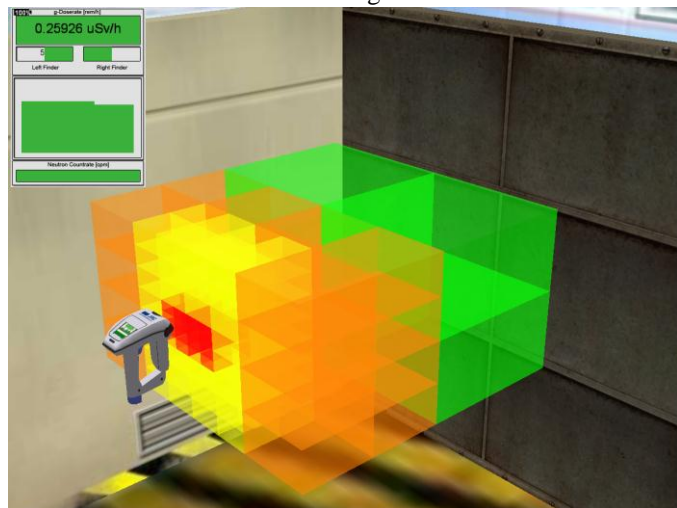


Fig. 16. Original source divided into 120 sub-sources (point mesh density). The four different levels of resolution are represented in different colours. Four green 1st level sources (biggest resolution), 24 orange 2nd level sources, 60 yellow 3rd level sources, and 32 red 4th level sources (smallest resolution).

Second, in order to test real-time behavior, execution time was measured for the Variable case using functionality supplied by Virtools. There is no need to monitor the Mono PK execution time, as it is constant (1 point for all cases) and very fast (3ms on the tested computer).

Table III shows execution times for all positions measured, and the number of points representing the source at that measuring point for the Variable non-regular method.

The best case corresponds to the situation where after checking solid angles, no new source generation is needed. The worst case corresponds to the situation where solid angle check is over the limit, and new source points need to be generated.

TABLE III
EXECUTION TIME AND MESH DENSITY (NUMBER OF POINTS) OF VR SIMULATION

Distance [cm]	Execution time [ms]	Execution time [ms]	Mesh density [source points]
	Best Case	Worst Case	
		Frontal	
10	216	380	120
20	110	184	64
30	62	96	36
50	16	24	8
		Top	
10	166	478	148
30	64	100	36
50	63	98	36
		Diagonal	
10	138	181	50
20	36	62	22
30	39	60	22
50	16	26	8

From the difference between best and worst case we can infer the time that each part of the method takes. The best case time corresponds to the time taken to make the PK calculus, and the difference between worst and best case is the time taken by the method to generate the new source model.

B. Case 2: Water Cube (¹³⁷Cs source) computer simulations

The second test considers a gamma radiation source composed of the isotope Cs 137 diluted in water. The shape of the source is a cube. The size of the cube is 20cm per side. The total activity of the source is 43.53 GBq. Only the 662 KeV energy line is considered with a yield of 84.6%.

Mass absorption coefficient and Build up factor for water are taken from [16], as well as the air absorption coefficient.

This second test compares dose rate results for a specific setup already used as a benchmark case in previous Dose rate experiments published in this journal [13].

The VR application is tested against the results provided by other PK non-VR software, one multi-PK code CIDEDEC, and a Mono PK code (NUCLEONICA).

MCNP Version 4C2 (Monte Carlo) code (simulating 10 million particles) is used as a benchmark of precision.

TABLE IV
DOSE RATE RECEIVED AT DETECTION POINT (AIR MEDIUM) [MSV/H]

Distance [cm]	MCNP	Nucleonica ^a	Dev ^b	CIDEC ^c	Dev ^b	VR prototype ^d	Dev ^b	VR 64 fix ^e	Dev ^b
500	0.10	0.12	20%	0.11	10%	0.115	15%	0.114	14%
100	2.63	2.89	10%	2.89	10%	2.889	10%	2.903	10%
50	11.1	11.6	5%	11.8	6%	11.55	4%	11.83	7%
30	32.8	32.2	2%	33.7	3%	33.21	1%	33.76	3%
15	141	128	9%	140	1%	138.9	1%	138.6	2%

^aNucleonica (Mono Point Kernel code), ^bDeviation to MCNP result, ^cCIDEC (Multi Point Kernel code regular mesh, unknown mesh density), ^dDeveloped VR application, ^eFixed 64 point regular mesh version of developed VR application.

Furthermore, in order to test the effect of the existing regularly spaced point meshes with respect to the developed non-regular mesh representation, a fixed 64 point regular mesh version is also tested in the same configuration. 64 points is chosen due to the fact that this is the maximum number of mesh points the developed application uses for this case (at the shortest distance measured).

The results of the dose rates calculated with the different computer codes and their deviations to the benchmark code (MCNP) are shown in tabulated form in Table IV.

To better appreciate the trends of these results, the deviations are shown on the following graph (Fig 17.)

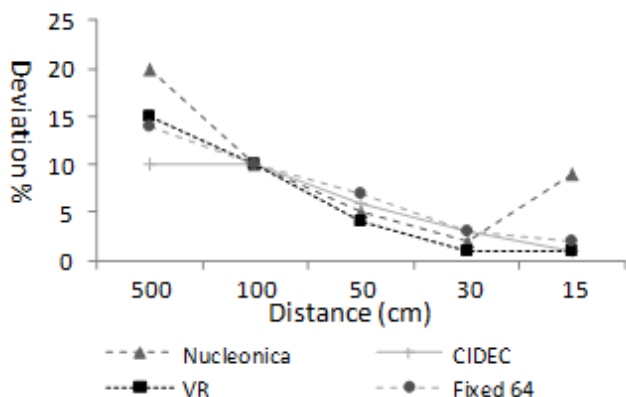


Fig. 17. Graph chart comparing the dose accuracy error of the developed variable method algorithm (VR) with respect to a Mono PK code (Nucleonica) and fixed Multi PK codes (CIDEC and Fixed 64).

The Mono PK code (Nucleonica) provides an acceptable accuracy in intermediate cases. But the deviation at close distances increases notably compared to the more detailed meshes as the graph shows.

The fixed regular mesh examples (CIDEC and 64 point VR) as expected reduce the accuracy error with respect to Mono PK at extremes but it is not providing any advantage in the intermediate cases, despite the extra computational effort.

The developed new VR non-regular version manages the best accuracy in all cases except the furthest distance, not only complying with requirements but also surpassing the alternatives.

Finally, in order to analyze the effect of the variable method with respect to fixed mesh methods, the total amount of time for all five measurements is calculated.

There is a clear proportional relationship between execution

time and number of points in the mesh. Two cases are considered for VR Variable code. The best case scenario when no change in source representation is required after the solid angle check; and the worst case scenario when a new source models needs to be generated for that case.

The PK computation expands through various frame-rendering loops as each individual point is calculated in one frame.

TABLE V
EXECUTION TIME CASE 2

Distance [cm]	Mono PK [ms]	VR Variable Best / Worst Case [ms]	VR Fix-Mesh 64 points [ms]
500	8	8 / 8	92
100	8	9 / 9	94
50	10	9 / 9	94
30	8	16 / 25	92
15	10	109 / 184	93
Total Time	44	151 / 235	465

Iteration and Cumulative execution time for VR application on fixed Mono PK, fixed 64-point regular mesh, and normal operation modes (on a computer with the following characteristics (Intel Xeon © E5640 CPU at 2.67GHz, Usable RAM 3.49 GB, NVIDIA Quadro FX 3800 graphics card, 32bit Windows 7 OS, Virtools 5.0 SDK).

V. CODE LIMITATIONS

A series of limitations are present in this prototype application.

- 1) Geometrical limitations: Geometrical source shapes are currently restricted to parallelepipeds. In practice this limitation might not be such as most cases are actually parallelepipeds (as shown in the result cases) but it can be a source of error when trying to measure large cylindrical or spherical sources at close distances.
- 2) Inherent PK method limitations:

--First, PK method uses a series of data tables that cover the [0.010MeV..30MeV] energy range for attenuation coefficients, [0.015MeV .. 15MeV] for Build Up factor coefficients, and [0.01MeV .. 20 MeV] for absorption coefficients. For energy lines outside the range of the tables, extrapolation is used instead of linear interpolation. In practice this is hardly a limitation due to the fact the apparatus being simulated have an operating range which is within the application's tables range (i.e. the Victoreen detector used in testing has an energy range of [0.023MeV .. 1.3 MeV].

--Second, Another limitation of the PK method is the amount of materials for which data exists. For compound

materials (like Case 1 KCl), new PK coefficients tables need to be created, based on averaging mass composition of the elements contained for which data is available, possibly incurring in an error.

- 3) Shielding limitations: Only one layer of shielding is currently considered (the closest to the source). There is only shielding data available for the restricted list of materials covered by the tables.
- 4) Flux limitation: Only direct and build up flux is considered, backscattering, and other secondary sources are disregarded.

VI. CONCLUSION

A method to estimate dose rates generated by gamma sources using the point kernel method has been developed. It uses a novel approach to handle a variable source representation in real time. This method has been used to develop a prototype VR based simulator application for training purposes.

The method has been tested in two cases: first, comparing against a real detector and second, comparing against other existing commercial software codes.

Three questions arise when trying to reach a conclusion.

--First, *Is this a valid method to simulate a handheld detector?* The results show that the developed method meets the accuracy requirement set (~20%)

Furthermore it represents an improvement with respect to the previously used mono PK methods in terms of accuracy.

In terms of execution time, even the worst-case measurement (0.31s) remains well below the specified response time (1.8s) of the detector. Therefore, the method successfully meets the requirements and positively answers the question.

--Second: *Is this detailed source representation necessary?* The results from table I show that the simpler Mono point kernel approach, albeit being very fast (3ms) fail to comply with accuracy requirements, they give in the worst case scenario, over four times as much (89%) accuracy error as the limit stated in the requirement. This justifies the need of a more detailed (multi-point) source representation than simple point kernel for this kind of simulation.

--Third: *Is the developed source representation (variable non-regular) better than the existing (fixed mesh) representations?* Again, the answer is "yes". The Graph in Fig. 14 indicates that the developed method results in a lower accuracy error than a regular mesh (for the same amount of points) in all measurements but one. Furthermore, the developed method provides a total execution time advantage compared to the fixed regular mesh as shown in Table V.

REFERENCES

- [1] T. Moltó Caracena, D. Brasset, E. Vendrell Vidal, E Ruiz Morales, J.G.M. Gonçalves, "A Design and Simulation Tool for Nuclear Safeguards Surveillance Systems," *ESARDA 31st annual meeting*, 2009.
- [2] T. Moltó Caracena, J.G.M. Gonçalves, P. Peerani, E. Vendrell Vidal, "Virtual Reality based Simulator for Dose Rate Calculations in Nuclear Safeguards and Security," *ESARDA symposium*, Budapest, 2011.
- [3] S. Stansfield, "Application of Virtual Reality to Nuclear Safeguards and Nonproliferation," *Sandia National Laboratories*, Albuquerque, NM USA, 1996.
- [4] IAEA, Department of Safeguards, "STR-368 International Target Values 2010 for measurement uncertainties in Safeguarding Nuclear Materials," Vienna November 2010.
- [5] L.E. Smith et al., "Coupling Deterministic and Monte Carlo Transport Methods for the Simulation of Gamma-Ray Spectroscopy Scenarios", *IEEE Trans. Nucl. Sci.*, vol. 55, No 5, October 2008.
- [6] G.A. Warren, L.E. Smith, M. Cooper, W. Kaye, "Evaluation Framework for Search Instruments", *IEEE Nuclear Science Symposium Conference record*, 2005
- [7] I. M. Prokhorets , S. I. Prokhorets , M. A. Khazhmuradov , E. V. Rudychev and D. V. Fedorchenko "Point Kernel method for radiation fields simulation", *Probl. Atom. Sci. Technol.*, vol. 48, pp.106 2007
- [8] A.A.C. Mol, C.A.F. Jorge, P.M Couto, S.C. Augusto, G.G. Cunha, L. Landau, "Virtual Environments simulation for dose assessment in nuclear plants", *Progress in Nuclear Energy.*, vol. 51, pp. 382, 2009.
- [9] S. M. Bowman, "SCALE 6: Comprehensive Nuclear Safety Analysis Code System,"*Nucl. Technol.* 174(2), 126-148, May 2011.
- [10] Y. Wu et al., "CAD-based interface programs for fusion neutron transport simulation," *Fusion Engineering and Design*, vol. 84, no. 7-11, pp. 1987-1992, June, 2009.
- [11] Nucleonica web driven nuclear science; *Dosimetry and shielding module*; Available: <http://www.nucleonica.net>; 2007-2011.
- [12] D.T. Ingersoll, "PUTZ: A Point-Kernel Photon Shielding Code", Oak Ridge National Laboratory, ORNL/TM-9803, 1986.
- [13] O. Vela, E. De Burgos, JM. Perez, "Dose rate assessment in complex geometries," *IEEE Trans. Nucl. Sci.* vol. 53, no 1. Feb 2006.
- [14] Y. Li, L. Lu, A. Ding, et al, "Benchmarking of MCAM 4.0 with the ITER 3D model," *Fusion Engineering and Design*, vol 82, pp. 2861-2866, 2007.
- [15] Dassault Systèmes. *3DVIA Virtools Version 5*. 1999-2008.
- [16] *Gamma-ray attenuation coefficients and buildup factors for engineering materials*, ANSI/ANS-6.4.3-1991, American Nuclear Society, 1991.
- [17] S. Bin Samat, S. Green, A.H. Beddoe, "The 40K activity of one gram of potassium" *Phys. Med. Biol.*, vol. 42 pp. 407-413. 1997.