Doctoral thesis on Biomedical Engineering
Universitat Politècnica de Catalunya

PyMCGPU-IR: a new tool for patient dose monitoring in interventional radiology procedures

David Fernández Bosman

Supervisor: Mercè Ginjaume Egido
Co-Supervisor: Maria Amor Duch Guillen

December 2023

Doctoral thesis on Biomedical Engineering
Universitat Politècnica de Catalunya

PyMCGPU-IR: a new tool for patient dose monitoring in interventional radiology procedures

David Fernández Bosman

Supervisor: Mercè Ginjaume Egido

Co-Supervisor: Maria Amor Duch Guillen

December 2023

Contents

Dedication

To my parents, who guide me in the art of learning.

Acknowledgement

I would like to express my gratitude to the Consejo de Seguridad Nuclear and the Cátedra Argos for their financial support, which has make the development of this thesis possible.

My deepest gratitude to Mercè Ginjaume and Maria Amor Duch for giving me the chance and for their patience throughout the completion of this project.

Thanks to all the people I have met at INTE. The enjoyable moments during lunch breaks significantly enhanced the overall experience of the journey. I want to extend my thanks to Victor and Anna Camp for their invaluable help throughout the thesis development. Additionally, I am grateful to Anna Torras for being a nice companion during our break times.

I would like to express my deepest gratefulness to my family, for encouraging and supporting me during the most challenging moments.

Finally, I could not have undertaken this journey without Crisitna, who has been there all the time, enduring my contradictions, and supporting me in every decision.

Abstract

Interventional radiology procedures are associated with potentially high radiation doses to the skin. The 2013/59/EURATOM Directive establishes that the equipment used for interventional radiology must have a device or a feature informing the practitioner of relevant parameters for assessing patient dose at the end of the procedure. Monte Carlo codes of radiation transport are considered to be one of the most reliable tools available to assess doses. However, they are usually too time consuming for use in clinical practice. This thesis has been developed at the Institute of Energy Technologies of the Universitat Politècnica de Catalunya within the framework of the European project "Implications of Medical Low Dose Radiation Exposure" (MEDIRAD). The main objective of this work is to develop a software tool based on the Monte Carlo program MC-GPU for assessing the skin dose in patients undergoing interventional radiology (IR) procedures. The achievement of this objective can be divided into two main blocks: the validation of MC-GPU and the development and validation of PyMCGPU-IR, a skin dose calculation tool for IR procedures based on MC-GPU. For the validation of MC-GPU, simulations were conducted and compared with the well-validated code PENELOPE/penEasy and then compared against thermoluminescent measurements performed on slab phantoms, both in a calibration laboratory and at a hospital. MC-GPU demonstrated excellent agreement in organ dose distribution, with differences below 1%, despite reducing the calculation time by a factor of 2500. Comparisons with thermoluminescent measurements indicated agreements within 10%, validating MC-GPU's ability to provide accurate dose estimates in real clinical setups in very short times. In this work we have also developed PyMCGPU-IR, a new software tool based on the Monte Carlo program MC-GPU for assessing skin dose and organ doses in patients undergoing an interventional radiology (IR) procedure. PyMCGPU-IR has been validated through skin and organ dose measurements in an anthropomorphic phantom and showed differences below 6% in skin dose measurements and mostly below 20% in organ doses in clinical procedures. PyMCGPU-IR offers both, high performance and accuracy for dose assessment when compared with skin and organ dose measurements. It also allows the calculation of dose values at specific positions and organs, the dose distribution and the location of the maximum dose per organ. In addition, PyMCGPU-IR overcomes the time limitations of CPU-based MC codes. In this thesis we have shown that PyMCGPU-IR is an innovative Skin Dose Calculation (SDC) tool that offers higher performance and accuracy for skin dose calculations compared to most available SDCs. Currently, PyMCGPU-IR provides dose values only after the procedure has finished. In the future, PyMCGPU-IR could be adapted to provide real-time dose calculation if real-time radiation source information is available.

Part I

Introduction

Chapter 1

Radiation protection in Interventional Radiology

Interventional radiology (IR) is a subspecialty within the field of radiology that uses X-ray radiation for image-guided diagnosis and treatment. The use of these techniques has yielded valuable results and has experienced significant growth in recent decades [1]. Fluoroscopy is a commonly employed modality that uses X-ray radiation to produce real-time images of a patient's internal structures. It is used in various fields, including interventional cardiology (IC).

During a fluoroscopic procedure, an X-ray beam is generated by an X-ray source and passes through the patient's body before reaching an image detector. This process generates a real-time image of the patient's interior, aiding the operator during the intervention. Most modern fluoroscope systems incorporate X-ray image intensifiers, which amplify the X-ray radiation into visible light at higher intensities compared to fluoroscope screens.

The advantages of fluoroscopic guidance procedures stem from their non-invasive nature, which is less traumatic than conventional treatment procedures like surgery. However, these benefits are accompanied by the risks associated with ionizing radiation. Real-time images are obtained by generating a series of consecutive single X-ray images over time, typically at a rate between 3 to 30 images per second, being the most frequent between 7.5 to 15 images per second. Although the radiation dose used for each individual image is very low, the cumulative dose can sometimes result in high skin exposure. The probability of sustaining an injury from radiation exposure is generally quite small. Ideally, the benefits of fluoroscopic procedures outweigh the radiation risks for the patient, which are typically much smaller than other non-radiation-related risks such as anesthesia or sedation.

The effects of radiation exposure can be categorized into two types: deterministic and stochastic effects. Deterministic effects occur when a threshold dose is surpassed, and their severity increases with the dose. In contrast, stochastic effects do not have a minimum threshold and may take decades to manifest. The probability of experiencing a stochastic effect increases with the total cumulative radiation energy over time, but the damage is independent of the dose [2]. In IR, the most common deterministic effect on patients is skin burns, which occur at doses higher than 2 Gy [3, 4, 5].

In 1994, the U.S. Food and Drug Administration (FDA) published a report containing recommendations aimed at preventing these injuries. It emphasized the inclusion of information necessary to estimate the skin dose in the patient's record [6]. In 2000, the International Commission on Radiological Protection (ICRP) published a set of recommendations to avoid the risk of radiation injuries [7]. For any procedure where the maximum cumulative absorbed dose approaches 3 Gy (or 1 Gy for repeatable procedures), the interventionist should document the estimated dose and the beam entry point in the patient's record. Furthermore, for patients exceeding these doses, a follow-up should be conducted for 10 to 14 days after the intervention to identify any effects on the skin that may result in more serious or chronic damage. ICRP 105 [8] also stressed the importance of real-time monitoring to detect whether the threshold doses for deterministic effects are being approached or surpassed for a particular patient. The absorbed dose in the skin at the area of maximum cumulative skin dose is the primary concern for fluoroscopically guided interventional procedures. Accordingly, the clinical protocol should include a statement on cumulative skin doses and skin sites.

The European Directive 2013/59/EURATOM [9] establishes the fundamental safety regulations for protecting against risks associated with ionizing radiation and recognizes interventional radiology (IR) as a specialized "practice" requiring particular attention to quality assurance programs and patient dose assessment. Article 60 specifies that all X-ray machines used in IR must provide the necessary parameters to evaluate the absorbed skin dose in patients.

The Peak Skin Dose (PSD) or Maximum Skin Dose (MSD) is the key parameter used to quantify the absorbed dose and was initially introduced by the Society of Interventional Radiology (SIR) and the European Society of Cardiovascular and Interventional Radiology (CIRSE) in their publication "Guidelines for patient radiation dose management" in 2009 [10]. The International Commission on Radiological Protection (ICRP) published recommendations for patient radiation protection in 2007 [11], which state that in Interventional Cardiology (IC), the PSD, along with the skin dose distribution, should be reported after the procedure. The most recent recommendations on Radiological Protection in international guidelines can be found in [12, 13].

This work is part of the European project "Implications of Medical Low Dose Radiation Exposure" (MEDIRAD) [14]. The primary objective of MEDIRAD was to enhance the scientific basis and clinical practices of radiological protection in the field of medicine. Achieving this goal requires a better understanding and evaluation of the effects of low-dose ionizing radiation resulting from diagnostic medical treatments and therapies. The project, funded by the European Commission, spanned four years and involved a large multidisciplinary European Consortium in the field of radiological protection, representing two European platforms: MELODI and EURADOS. It is based on three major objectives:

1. Improvement of organ dose estimation and management in order to a) optimize doses and set recommendations, and b) to provide adequate dosimetry for clinical-epidemiological studies of effects of medical radiation.
2. Evaluation of the effects of medical exposure, focusing on the two major endpoints of public health and clinical relevance:
    a Short-term cardiovascular effects of low to moderate doses of radiation, including underlying mechanisms.
    b Long-term effects of low to moderate doses on cancer risk.
3. Building upon the scientific findings and fostering collaboration between the medical and nuclear scientific communities to develop science-based policy recommendations for decision-makers and practitioners, ensuring effective protection of patients, workers, and the general public.

To achieve these objectives, the project was divided into six working packages (WP). A research group from the Universitat Politècnica de Catalunya (UPC) participated in WP2, specifically in subtask 2.2.2. The objective was to develop a real-time patient dose monitoring tool for fluoroscopically guided interventional procedures. The ICRP publication in 2013 [15] predicted that "in the near future, it may be possible to obtain skin dose estimates and skin dose maps in real-time using automated methods." Thus, the current challenge is to develop online (real-time) Skin Dose Calculation (SDC) software tools based on Monte Carlo simulation of radiation transport.

Chapter 2

Methods for estimating skin dose

Ideally, one would like to quantify and visualize both the Peak Skin Dose (PSD) and skin dose distribution in real time to help the interventionist reduce the risk of deterministic effects. However, calculating the skin dose is complex, and currently, real-time calculation of these data is very challenging. There are two main approaches used to assess the skin dose in interventional radiology (IR) procedures: (1) dosimetry measurements and (2) dose calculation methods.

2.1 Dosimetry measurements

This approach involves estimating the skin dose based on dose measurements. The dose measurements can be classified as (a) direct dose measurements or (b) indirect dose measurements:

2.1.1 Direct dose measurements

Direct dose measurements involve placing detectors on the patient's skin to measure the dose. Various types of detectors have been used for dose measurements in clinical scenarios [16].

a Thermoluminescent detectors

Thermoluminescent detectors (TLDs) are commonly used for patient dosimetry. Some TLDs have good dosimetry performance and are "nearly tissue equivalent." TLDs also offer good spatial resolution accuracy due to their small size. Placing a TLD exactly at the position of the peak skin dose would provide a good estimate of the dose. However, in practice, it is challenging to predict the exact location of the PSD due to the complex geometry involved in IR procedures, where the patient may move with respect to the beam source multiple times, and the beam source can rotate in different planes. To address this limitation, multiple TLD detectors can be arranged in an array or grid to provide a skin dose mapping [17]. However, this approach can be cumbersome for practitioners to perform for each individual IR procedure. Additionally, the information from TLDs needs to be read out, so the results are not available in real time but are only delivered after hours or even days.

Other types of passive detectors, such as OSL, are also begining to be used [18], with similar advantages and limitations than TLDs.

b MOSFET detector

Metal-Oxide-Semiconductor Field-Effect Transistor (MOSFET) detectors are an alternative to TLDs. The main advantage of MOSFET detectors over TLDs is that they can provide real-time alerts when a certain threshold dose set by the operator is reached. However, the challenge of unknown PSD location still remains, making them unsuitable for routine skin dose assessment in clinical practice [16].

c Radiochromic films

Radiochromic film detectors contain a dye that changes color when exposed to ionizing radiation. XR-Type R Gafchromic films have been shown to be an efficient solution for measuring the patient's skin dose in IR procedures [19]. They are made of material that is nearly tissue equivalent and can cover a large area. This enables measuring the skin dose distribution with better spatial resolution compared to an array/grid of TLDs. Moreover, the PSD can be measured with better spatial accuracy as the film can capture the exact location with a higher probability than an array/grid of TLDs. However, in order to quantify the absorbed doses, films require careful calibration and also a control of the constancy of its response. Among its main advantages, its good spatial resolution can be highlighted. The price together with the time required for analyzing the radiochromic films reduces its use in practice.

d Radiosensitive detectors

Radiosensitive detectors are another type of point detector that changes color when exposed to X-rays. The absorbed dose can be determined by analyzing the color using color-measuring instruments. These detectors have a rectangular shape (1.3 cm x 1.8 cm) and can be arranged in rows and columns with a 5 cm separation for skin dose measurements. One advantage is that the absorbed dose can also be estimated visually by comparing the indicators with color samples arranged in increasing dose order. This allows the operator to estimate the skin dose immediately after the procedure [20, 21]. However, the dose distribution information is difficult, likewise with the other point detectors.

2.1.2 Indirect dose measurements

Another alternative is to define a set of indicators that can alert the operator in real-time about possible skin injuries due to deterministic effects based on readily available (online) dosimetric quantities in the angiographic system.

In the year 2000, the International Electrotechnical Commission introduced the reference point air kerma ($K_{a,r}$) [22], which was later adopted as part of the reference FDA performance standard for medical X-ray fluoroscopy systems [23]. $K_{a,r}$, also known as the reference dose or cumulative dose at the reference point, represents the total air kerma at a specific point in space. Typically, this reference point is located centrally in the direction of the X-ray beam and 15 cm below the isocenter for C-arm fluoroscopic systems. Another dosimetric quantity is the kerma-area product (KAP), which represents the total air kerma integrated over the cross-sectional area of the X-ray beam. These dose quantities can be correlated with the Peak Skin Dose (PSD) [24] to define dose indicator levels. Both KAP and $K_{a,r}$ have been shown to have a strong correlation with the PSD and, therefore, it is feasible to use them as dose indicator levels.

It is important to note that the PSD depends on various factors such as beam angulations, field size, patient position relative to the X-ray source, beam quality, table backscatter, and others. Therefore, relying solely on $K_{a,r}$ and KAP is not sufficient to calculate the PSD [25]. For instance, $K_{a,r}$ does not account for changes in projections during the procedure, resulting in an overestimation of the maximum incident air kerma at any point on the patient. Similarly, KAP, being the product of air kerma and field size, may yield the same KAP value for a small radiation dose in a large area and a large radiation dose in a small area. The latter scenario poses a greater risk to the patient. While both dosimetric indicators can be used together, the estimated PSD remains inadequate. Nevertheless, the use of $K_{a,r}$ is recommended over KAP [26].

Furthermore, the significant variation in the complexity of different types of interventional radiology procedures makes it impossible to define indicator levels that are independent of the hospital. In some IR procedures, only a few generic, hospital-independent dose indicator alerts are feasible. One potential solution is to define hospital-specific dose indicator alerts based on laborious experiments involving TLDs or Gafchromic film detectors.

2.2 Dose calculation methods

An alternative to estimate the PSD involves the use of calculation methods. Various software solutions, referred to as Skin Dose Calculation (SDC) tools, have been developed for this purpose. These

SDC tools can be categorized into two main groups: online and offline. Online SDC tools use live data streaming from the angiographic system to calculate the dose directly during the interventional procedure, enabling real-time delivery of dose information to the doctor. On the other hand, offline SDC tools use data stored in the Radiation Dose Structured Report (RDSR) to calculate the dose after the procedure has been completed, providing post-procedure dose information.

Some vendors have incorporated online SDC tools into their angiographic systems to calculate the PSD and, in some cases, other quantities such as 2D or 3D maps of the skin dose distribution of the patient. Offline SDC tools have been developed by research-based institutions or for commercial purposes, often as stand-alone products or as software packages integrated into the Dose Archiving and Communication System (DACS). There is a wide variety of SDC tools available. In this context, the recently completed European project VERIDIC (Validation and Estimation of Radiation skIn Dose in Interventional Cardiology) [27] aimed to review and validate existing SDC software products. Table 1 presents a list of the most relevant and frequently used SDC tools (according to [28]), which will be discussed in this work.

Table 1: Skin Dose Calculation (SCD) tools [28]

SDC tool name	Manufacturer
Dose Map	GE
DTS	CANON
em.dose	ESPRIMED
Dose	QAELUM
RDM	MEDSQUARE

The SDC tools listed in Table 1 adopt a similar approach for calculating the skin dose, which is based on the formalism proposed by Jones and Pasciak in 2011 [29]. This approach involves correcting the $K_{a,r}$ (reference point air kerma) displayed by the X-ray systems using a series of factors that account for various physical phenomena. These factors include the calibration of the displayed $K_{a,r}$ through quality control measurements, attenuation caused by the table and mattress, backscatter radiation, the distance between the reference point and the patient entrance point, and the conversion of air kerma to dose in tissue.

Equation 2.1 represents the mathematical expression used to calculate the skin dose (SD).

$$SD = k_{a,r} \cdot CF \cdot AC \cdot B \cdot \left(\frac{d_{ref}}{d_{perp}}\right)^2 \cdot f_{skin} \tag{2.1}$$

where CF represents the calibration factor, which is the ratio between the measured and displayed $K_{a,r}$. AC denotes the attenuation coefficient of the table and mattress, B represents the backscatter factor, d_{ref} is the distance between the X-ray focal spot and the reference point, d_{perp} is the distance between the X-ray focal spot and the patient entrance reference point ($PERP$), and f_{skin} is the ratio of the mass-energy-absorption coefficients from skin-to-air.

The SDC tools differ in their methodology for calculating the values of the factors in Equation 2.1. Here are the approaches used by some of the SDC tools.

In the Dose Map system, the backscatter factor is calculated using an algorithmic based on the incident beam area. It is a logarithmic model determined and calibrated through measurements performed in a water phantom on a GE HC X-Ray angiographic system [30, 31]. Another study [32] uses a different method for estimating the backscatter factor. In this case, the estimation is obtained through measurements using (TLDs) in an anthropomorphic thorax phantom on a GE Innova IGS 520 imaging system.

The Dose Track System (DTS) employs a predefined data file that tabulates the backscatter factor values for different system parameters such as kVp and beam filtration [33]. This data file is determined by conducting measurements with a calibrated ionization chamber placed at the center of the beam on the patient tabletop, using a phantom made of solid water material on a Toshiba Infinix C-arm unit.

The em.dose [34, 35], Dose [36], and RDM [37] systems, on the other hand, use backscatter factors interpolated from the coefficients obtained through the simulation work of Benmakhlouf et al [38]. These coefficients are provided as a function of tube kVp, inherent and additional filtration, and field size (considering square fields).

Each of these methods is used by the respective SDC tools to estimate the backscatter factor in their calculations.

To estimate the attenuation coefficients of the table and mattress, different models are employed by various SDC tools. The Dose Map system [31] uses a second-order polynomial equation to determine the attenuation coefficient as a function of the beam energy and beam filtration. This equation is derived by conducting measurements using an ionization chamber on a GE HC X-Ray angiographic system.

The DTS system, for the backscatter factor, employs its own tabulated data file that includes the attenuation coefficient for different system parameter values, such as beam angulation. The data file is generated by performing measurements with a calibrated ionization chamber placed on the patient tabletop under a phantom made of solid water material, specifically on a Toshiba Infinix C-arm unit.

In the em.dose system, the attenuation coefficient value is selected for each event based on previous measurements. This is done for each available beam filtration at three different beam energies and different beam angulations for an Allura Xper FD 10 unit [34].

The RDM system employs user-defined transmission coefficients, which are expected to be provided for Posterior-Anterior (PA) projections. The introduced value is automatically adjusted for different beam angulations. It should be noted that each angiographic system may have different transmission coefficients due to variations in the thicknesses and material composition of the table and mattress. The use of user-defined transmission coefficients in RDM ensures vendor independence.

According to the IEC standards (2000), the accepted tolerance for the displayed $K_{a,r}$ is 35%. This means that if the skin dose calculation is based solely on the $K_{a,r}$ without using a calibration factor (CF), the uncertainty of the skin dose calculation cannot be lower than the uncertainty associated with the $K_{a,r}$, which can be up to 35% in certain cases. Therefore, it is crucial to calibrate the displayed $K_{a,r}$ by determining and calculating the CF.

However, it should be noted that some vendors, like the Dose Map system, do not use the calibration factor CF, while others, such as the Dose system, do not specify the value used. The em.dose system employs calculated quality control factors specific to each clinical protocol [34]. In the case of the RDM system, users can indicate only one equipment-specific CF value [37].

Regarding the mass-energy-absorption coefficients from skin-to-air (f_{skin}), the Dose Map system does not apply any factor, and for the Dose system, it remains unclear whether an f_{skin} factor is used or not. Both the DTS [33] and em.dose [34] systems use a single value for the f_{skin} factor. Lastly, the RDM system employs backscatter factors interpolated from the coefficients provided by Benmakhlouf et al [38].

The SDC tools also exhibit differences in the patient models employed for calculating the skin dose. The simplest approach involves using super-elliptical models, which can be found in systems like Dose Map [31], em.dose [34], and Dose [36]. These models may or may not be scalable based on the real patient's height or weight.

More sophisticated models are also used. For instance, the RDM system [37] incorporates a parallelepiped rectangular geometry connected on both sides by two half-cylindrical geometries. By using isotropic 3D contour expansions, the size of the phantom can be adjusted to match the patient's morphology.

Another example is the DTS system [39], which has developed 3D humanoid graphics with varying weights and heights. These graphics are morphed to correspond to different body types using the open-source software application Makehuman [40]. In addition to the presented SDC tools, there are other existing SDC tools that generally follow a similar approach to calculate the PSD. The accuracy of these tools relies on the precision of the methodologies employed to determine the various factors in Equation 2.1.

Most SDC tools available are offline systems that extract the required data for PSD calculation from radiation dose structured reports (RDSR). Only the DTS and Dose Map tools are online systems that provide real-time PSD and dose mapping by directly extracting the necessary data from the modality. From a radiological protection perspective, online systems are preferable as they allow operators to visualize the irradiated areas of the patient's body and potentially reduce the PSD by avoiding those areas. However, collecting data from the modality can be complex and specific to each system. On the other hand, RDSR data should adhere to DICOM standards, making it accessible to all software developers.

The SDC tools also differ in terms of their output data and dose reporting. These differences include the dosimetric information displayed, such as PSD, skin dose map, dose map resolution, number of events, and $K_{a,r}$, among others. The graphical representation of the skin dose mapping also varies among the tools. Some tools use colour-coded dose maps painted on a 3D representation of the patient's body, while others use colour-coded paint on a 2D representation of the patient's back. For instance, the DTS system employs a colour-coded paint on a 3D representation, while the em.dose, Dose, and RDM systems use a colour-coded paint on a 2D representation. The Dose Map system, on the other hand, uses a grey-scaling paint on a 2D flat representation of the patient's back.

Another potential approach to assess skin dose in interventional procedures involves the development of a dosimetry tool based on Monte Carlo (MC) simulations, for which the main challenge is to obtain results in relatively short computation times. MC codes, which are considered the gold standard for radiation dose calculations, are generally time-consuming and do not provide quick estimations of the desired output. We present here the PyMCGPU-IR program [41]. PyMCGPU-IR is a Python application that automatically acquires X-ray source conditions from the patient's RDSR and is based on the MC-GPU code [42]. The MC-GPU code is freely distributed and implements a massively multi-threaded MC simulation algorithm for X-ray transport in voxel geometries, primarily used in diagnostic imaging.

There are only a few available MC codes capable of providing complex dosimetric results in short computation times. Examples include FDEIR, ARCHER, and irtGPUMCD [43, 44, 45]. These codes use the computational power of Graphics Processing Units (GPUs). FDEIR, based on MC-GPU, focuses on calculating skin doses in interventional radiology using voxelized phantoms made of air, water and bone, while ARCHER is designed for dose assessment in CT imaging and radiotherapy. Additionally, irtGPUMCD is used for dose calculation in nuclear medicine.

Part II

Aims

Chapter 3

Aims and outline

This thesis has been developed at the Institute of Energy Technologies of the Universitat Politècnica de Catalunya within the framework of the European project "Implications of Medical Low Dose Radiation Exposure" (MEDIRAD).

The main objective of this work is to develop a software tool based on the Monte Carlo program MC-GPU for assessing the skin dose in patients undergoing interventional radiology (IR) procedures. The achievement of this objective can be divided into two main blocks: the validation of MC-GPU and the development and validation of PyMCGPU-IR, a skin dose calculation tool for IR procedures based on MC-GPU.

1. Validation of MC-GPU

   MC-GPU is a Monte Carlo program used for X-ray transport simulation. Its validation involved, on a first step, comparing MC-GPU with a well-validated MC simulation code. In a second step, benchmarking the simulations against clinical measurements. The measurements were conducted at the UPC Calibration Laboratory under controlled conditions, as well as in a realistic environment at the Hemodynamic Department of the Hospital Universitari Vall d'Hebron (HUVH).

2. Development and validation of PyMCGPU-IR

   For the development and validation of PyMCGPU-IR, the main objectives were to enhance MC-GPU to meet the requirements for developing a computational skin dose monitoring tool and to validate PyMCGPU-IR against organ dose measurements in realistic clinical setups in IR. Measurements performed at the Centre Hospitalier Universitaire de Liège in Belgium, in collaboration with the Belgian Nuclear Research Centre (SCK CEN), were used for this purpose.

The thesis is structured into four parts. Part I, chapter 1 discusses the risks associated with ionizing radiation in IR and the increasing interest in developing new methods for monitoring patient dose. Chapter 2 provides an overview of available SDC tools and Chapter 3 in Part II outlines the aims of the thesis and the report's structure.

Part III presents the methods used for the simulations in three separate chapters. Chapter 4 introduces basic concepts related to MC simulations, GPU programming (CUDA), and IR procedures. It presents MC-GPU and PyMCGPU-IR, along with other subsections on simulations, such as X-ray spectrum generation. The uncertainty calculation methodology used in this work is also explained. Chapter 5 describes the dose measurements conducted for the validations of MC-GPU and the development and validation of PyMCGPU-IR. Finally, Chapter 6 presents the computational resources used in the study.

Part IV includes specific sections dedicated to the main results of the work, most of which have been presented in scientific meetings and published in peer-reviewed journals. Part V provides a discussion of the results, including limitations and suggestions for future work. Finally, Part VI summarizes the main conclusions of the thesis.

Part III

Methods

Chapter 4

Simulations

4.1 Basic concepts

In this chapter, we explain the methods used to perform the simulations. As stated in the objectives section, this work has two main goals: the validation of the MC-GPU program and the development and validation of PyMCGPU-IR, a software tool for calculating skin dose in interventional radiology (IR) procedures.

From these objectives, it is evident that this work encompasses three key concepts: Monte Carlo (MC) simulations, GPU programming (CUDA), and IR procedures. Below, we will provide an explanation of each concept.

4.1.1 MC method

The Monte Carlo (MC) method is a numerical technique that uses random sampling to solve various problems. It finds extensive application in solving complex physical and mathematical problems with multiple independent variables. MC simulations are particularly useful in simulating radiation transport physics.

In MC simulations of radiation transport, the trajectory of each particle is individually simulated, and its path or history is determined by a sequence of connected free flights. During these flights, the particle undergoes interaction events that can alter its energy and direction. The possible interactions are governed by an interaction model, which consists of a set of differential cross sections (DCS) defining the probability distribution functions (PDF) for the various random variables characterizing the particle's history. The results are obtained by averaging over all simulated histories, and increasing the number of simulated histories reduces the associated statistical uncertainty.

In the context of photon transport, the primary interaction processes with matter include Rayleigh scattering, photoelectric absorption, Compton scattering, and pair production. The DCS of each interaction process determines the probability of occurrence.

While the transport of photons in biological tissue can also be modeled using deterministic methods, such as the radiative transfer equation (RTE), analytical solutions to the RTE are often challenging or impractical, especially for complex geometries. Another alternative for solving the (RTE) is to use numerical methods.

MC simulations offer several advantages over numerical methods. Firstly, the accuracy of the simulations primarily depends on the accuracy of the adopted DCS. Secondly, MC simulations are a good candidate at tracking particles through complex material systems with intricate geometries, where deterministic methods may struggle to define appropriate boundary conditions. Additionally, evaluating statistical uncertainties in MC simulations is simpler compared to other numerical methods. MC simulations also allow for the simultaneous simulation of multiple variables with desired spatial and temporal resolutions. These advantages establish MC methods as the preferred approach for simulating photon transport in numerous biomedical applications. However, it is important to note that MC simulations can be computationally time-consuming compared to alternative numerical methods, especially when dealing with complex geometries and a large number of particles.

The accuracy of MC simulation results directly relates to the number of photons (histories) simulated. Increasing the number of histories reduces the associated statistical uncertainty and improves the precision of the simulated results. However, larger numbers of simulated histories correspond to longer computational times. Thus, finding an optimal balance between precision and computational time is crucial in MC simulations.

4.1.2 CUDA

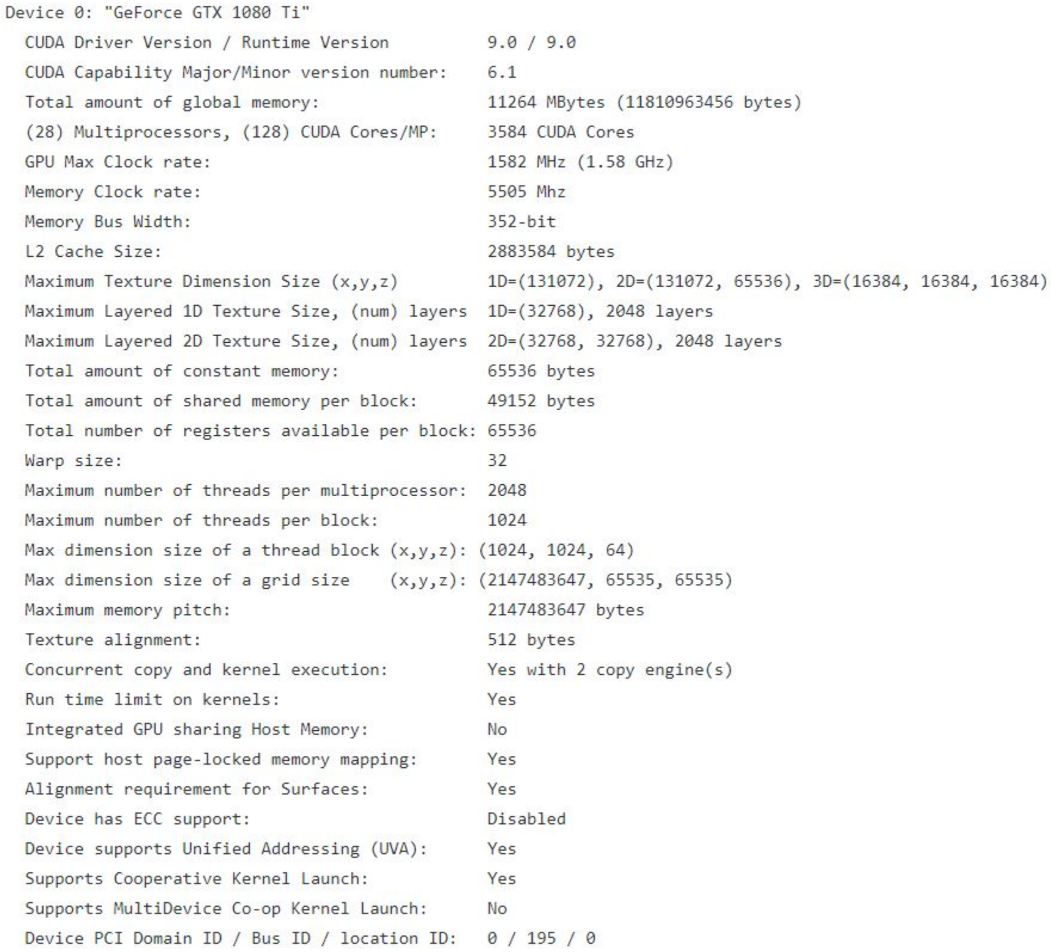
CUDA stands for Compute Unified Device Architecture and was developed by Nvidia [46]. It is a parallel computing platform that uses Graphic Processing Units (GPUs) for processing and executing operations instead of relying solely on the Central Processing Unit (CPU).

From a hardware perspective, a GPU consists of streaming multiprocessors (SMs). Each SM contains execution processing units known as CUDA cores (which can be thought of as analogous to CPU cores). The number of SMs and CUDA cores per SM varies depending on the specific GPU model.

For example, the GeForce GTX 1080 Ti GPU has 28 SMs and 128 CUDA cores per SM (see Figure 1).

At the programming level, tasks are parallelized and distributed across the GPU using blocks and threads, which are programming abstractions. Blocks consist of threads, and each thread contains the sequence of operations to be executed on a single CUDA core. Blocks are distributed among SMs, and threads within a block are organized into Warps, with each Warp consisting of 32 threads. A visual representation of the CUDA architecture is shown in Figure 2.

While CPUs are designed to execute threads quickly, they can only handle a limited number of threads in parallel. In contrast, GPUs can handle thousands of threads in parallel. CPUs demonstrate better performance for single-thread executions compared to GPUs. However, the GPU's lower performance in individual thread execution is offset by the significant computational gains achieved through massive parallelization, especially in terms of computational time performance.



```
Device 0: "GeForce GTX 1080 Ti"
  CUDA Driver Version / Runtime Version          9.0 / 9.0
  CUDA Capability Major/Minor version number:    6.1
  Total amount of global memory:                 11264 MBytes (11810963456 bytes)
  (28) Multiprocessors, (128) CUDA Cores/MP:     3584 CUDA Cores
  GPU Max Clock rate:                            1582 MHz (1.58 GHz)
  Memory Clock rate:                             5505 Mhz
  Memory Bus Width:                              352-bit
  L2 Cache Size:                                 2883584 bytes
  Maximum Texture Dimension Size (x,y,z)         1D=(131072), 2D=(131072, 65536), 3D=(16384, 16384, 16384)
  Maximum Layered 1D Texture Size, (num) layers  1D=(32768), 2048 layers
  Maximum Layered 2D Texture Size, (num) layers  2D=(32768, 32768), 2048 layers
  Total amount of constant memory:               65536 bytes
  Total amount of shared memory per block:       49152 bytes
  Total number of registers available per block: 65536
  Warp size:                                     32
  Maximum number of threads per multiprocessor:  2048
  Maximum number of threads per block:           1024
  Max dimension size of a thread block (x,y,z): (1024, 1024, 64)
  Max dimension size of a grid size    (x,y,z): (2147483647, 65535, 65535)
  Maximum memory pitch:                          2147483647 bytes
  Texture alignment:                             512 bytes
  Concurrent copy and kernel execution:          Yes with 2 copy engine(s)
  Run time limit on kernels:                     Yes
  Integrated GPU sharing Host Memory:            No
  Support host page-locked memory mapping:       Yes
  Alignment requirement for Surfaces:            Yes
  Device has ECC support:                        Disabled
  Device supports Unified Addressing (UVA):      Yes
  Supports Cooperative Kernel Launch:            Yes
  Supports MultiDevice Co-op Kernel Launch:      No
  Device PCI Domain ID / Bus ID / location ID:   0 / 195 / 0
```

Figure 1: Features of the GPU model GeForce GTX 1080 Ti.

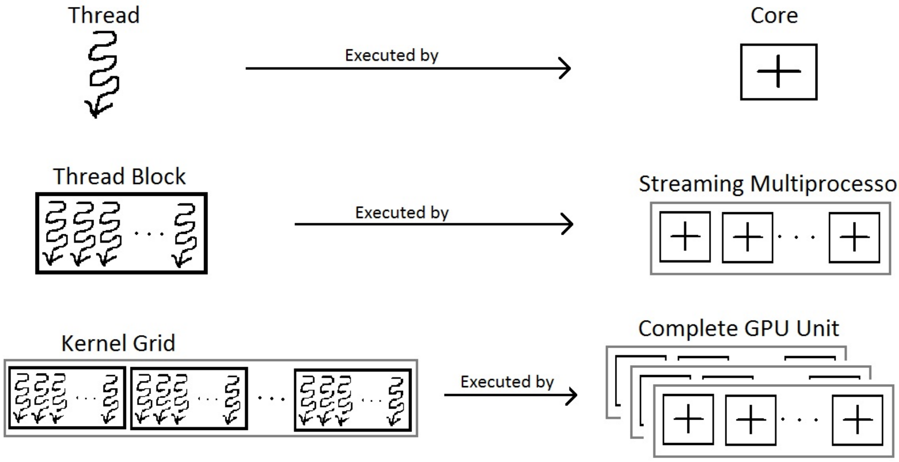


Figure 2: Scheme of the CUDA architecture. Image extracted from [47]

4.1.3 Interventional Radiology (IR) procedure

As explained in Chapter 1, Interventional Radiology (IR) is a specialized field within radiology that uses X-ray radiation for image-guided diagnosis or minimally invasive treatments. By using real-time imaging, IR operators can access internal structures of the patient using small needles and wires, thereby avoiding the need for surgical procedures, reducing pain, and minimizing recovery time. Fluoroscopy, which generates real-time X-ray images of internal structures, is the most commonly employed modality in IR.

An IR equipment typically consists of two primary components. Firstly, there is an X-ray generator and an X-ray tube where the X-rays are produced. Secondly, there is an image intensifier or a flat-panel detector where the X-ray beam, after passing through the patient, is detected to form an image. Different types of fluoroscopy machines with various components and configurations exist depending on the specific intervention being performed. The experimental measurements conducted in clinical scenarios for this work were carried out using a C-arm. C-arms are the most prevalent type of fluoroscopy machines. Figure 3 illustrates a C-arm machine, which connects the X-ray generator and the image intensifier or flat-panel detector. A flat table is positioned between the X-ray generator and the image intensifier or flat-panel detector, serving as a platform on which the patient lies during the procedure. The C-arm can rotate in two different planes, as described in detail in section 4.7. Additionally, the C-arm can also move horizontally and vertically. This allows irradiation of each region of the patient from multiple angles, generating different images with diverse perspectives of the targeted region.

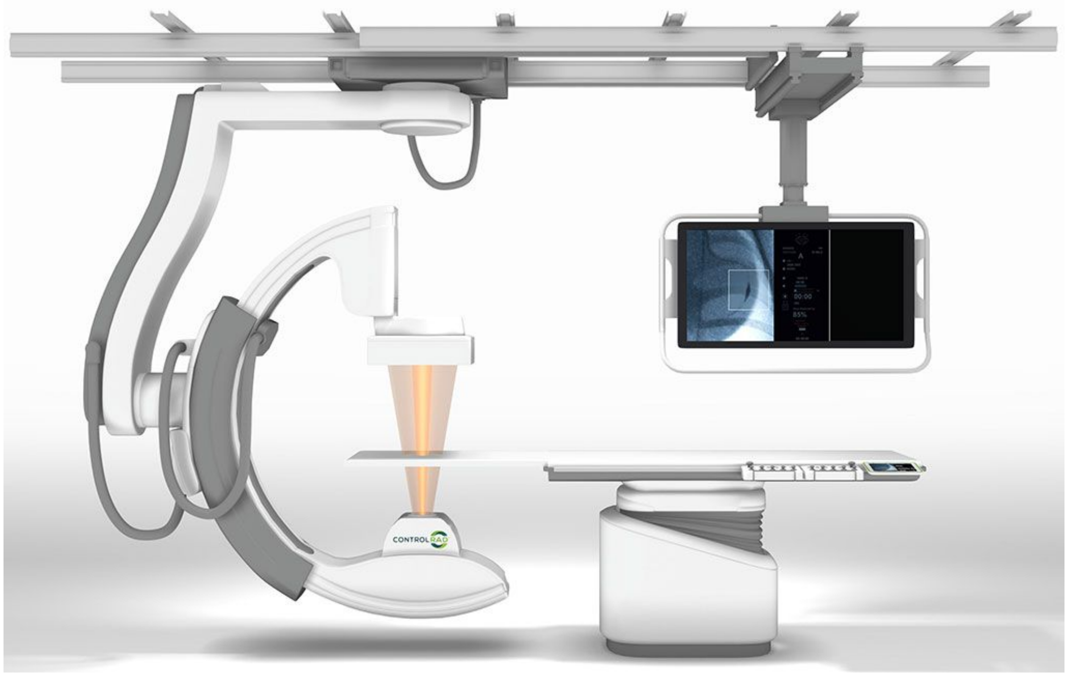


Figure 3: C-arm machine. Image extracted from [48]

All images and related information pertaining to the irradiations performed during the procedure are stored using the Digital Imaging and Communications in Medicine (DICOM®) protocol [49]. DICOM® is an international standard for medical images and associated information. It defines formats for medical images that can be exchanged with the required data and quality for clinical use. Information regarding the radiation dose is stored in the Radiation Dose Structured Report (RDSR), which adheres to the DICOM standards. Most existing Skin Dose Calculation (SDC) tools extract the necessary data from RDSR to calculate the patient skin dose.

4.1.4 Voxelized phantoms

The human anatomy in MC-GPU is represented using voxelized geometries. A voxelized geometry is a model that uses small volumes called voxels to represent the volumetric shape of an object, such as the human body. Each voxel has a rectangular shape and is placed adjacent to neighboring voxels, ensuring that their faces are in contact, forming a continuous shape. In this arrangement, all voxels except those on the contour of the object (known as bounding voxels) have six faces touching six neighboring voxels. The voxelized geometries used in MC-GPU adopt the PENELOPE/PenEasy format [50]. All voxels have the same dimensions and are composed of a homogeneous material with specified density.

For this work, various anthropomorphic phantoms were employed. Anthropomorphic phantoms are objects designed to simulate the human body, specifically patients. They are composed of materials with tissue characteristics similar to those of biological organisms and can be either physical or computational (mathematical models). Anthropomorphic phantoms serve multiple purposes due to their resemblance to real patients. For instance, certain physical anthropomorphic phantoms feature small cavities where TLD detectors can be placed to measure organ doses, while computational phantoms are used to simulate patient geometry in Monte Carlo dose calculations.

a Duke computational phantom

The Duke anthropomorphic phantom represents a 34-year-old male and is part of the Virtual family, which consists of four whole-body models [51]. The Duke voxel phantom comprises 122 x 62 x 372 voxels with dimensions of 0.5 x 0.5 x 0.5 cm^3. In the simulations, nine materials were considered from the initially available segmented tissues: muscle, soft tissue, bone, cartilage, adipose tissue, blood, skin, lung, and air.

b Rando Alderson phantom

An adult Rando Alderson (RA) anthropomorphic phantom [52] was used in the experiments. The RA phantom represents a patient weighing approximately 75 kg and is horizontally sectioned into 36 slices with a thickness of 2.5 cm. Each slice consists of soft, bone, or lung-equivalent tissue and contains small cavities for placing TLD detectors. The voxel model of the RA phantom was derived from CT images of the phantom used in the measurements [53]. The voxelized RA phantom comprises 256 x 361 x 185 voxels with dimensions of 0.15 x 0.1 x 0.5 cm^3. The small cavities for TLD detectors have a volume of 0.11 cm^3 and are considered to be filled with air for the simulations.

c REX and REGINA computational phantoms

The REX and REGINA phantoms are adult reference computational phantoms for male and female patients, respectively. The voxel models for the REX and REGINA phantoms used in this work were developed by Helmholtz Zentrum München [54, 55]. The Rex and Regina phantoms have the same anatomy but a different elemental tissue composition compared to the ICRP Publication 110 reference phantoms [55]. In this work, the original 145 segmented objects in the phantoms were simplified and grouped into 27 different materials. The Rex voxel dimensions are 0.21 x 0.21 x 0.80 cm^3, resulting in a voxel volume of 36.53 mm^3, while the Regina voxel dimensions are 0.18 x 0.18 x 0.48 cm^3, resulting in a voxel volume of 15.25 mm^3.

d Other voxelized geometries

Other self-made voxelized geometries have been used in the simulations and are explained in the corresponding sections in Part IV.

4.2 MC-GPU

MC-GPU is a Monte Carlo simulation code designed to generate synthetic radiographic images of realistic models of the human anatomy using the computational power of commodity Graphics Processing Unit (GPU) cards. The code uses a massively multithreaded Monte Carlo simulation algorithm to transport X-rays in a voxelized geometry.

The initial version of the code was developed by Badal et al. in 2009 [42] using the CUDA 2.2 programming model. The X-ray interaction models and material properties were adapted from the PENELOPE 2006 code [56]. The code was optimized to improve performance when utilizing multiple GPUs, implementing efficient interaction sampling and geometry ray-tracing algorithms.

MC-GPU enables the execution of thousands of sub-processes in parallel on GPUs. The source code and auxiliary files of the 2009 version of MC-GPU were released as free and open software in the public domain in 2010 and can be obtained from the website: `http://code.google.com/`. In 2012, MC-GPU version 1.3 was updated to CUDA 5, but no changes were made to the physics models. The physics models were updated to the PENELOPE 2014 version [50] for this work. Notably, the main change between the 2014 and 2006 versions of PENELOPE was the implementation of new cross sections for Rayleigh and photoelectric effects [57]. While differences in total mean free paths are small at low energies (2-3% depending on the material), updating the materials description is relevant for this study due to the predominance of photoelectric interactions in the typical energy range of interventional radiology.

To enhance the capabilities of MC-GPU, the libraries were modified to increase the maximum number of different materials that could be considered in the calculations. The original maximum was set to 15, but it was increased to 30 in this work. It was determined that with the computational resources described in Chapter 6, simulations with up to 70 different materials could be performed. The impact on simulation efficiency, as reported by Saidi [58], was found to be negligible, lower than 1%.

In this section, the features of MC-GPU are presented, and important information regarding the operation of the code is described. The general workflow outline of MC-GPU is depicted in Figure 4. According to Figure 4, the MC-GPU program can be divided into three main blocks: the input block, the program block, and the output block. Let's explore each of these blocks.

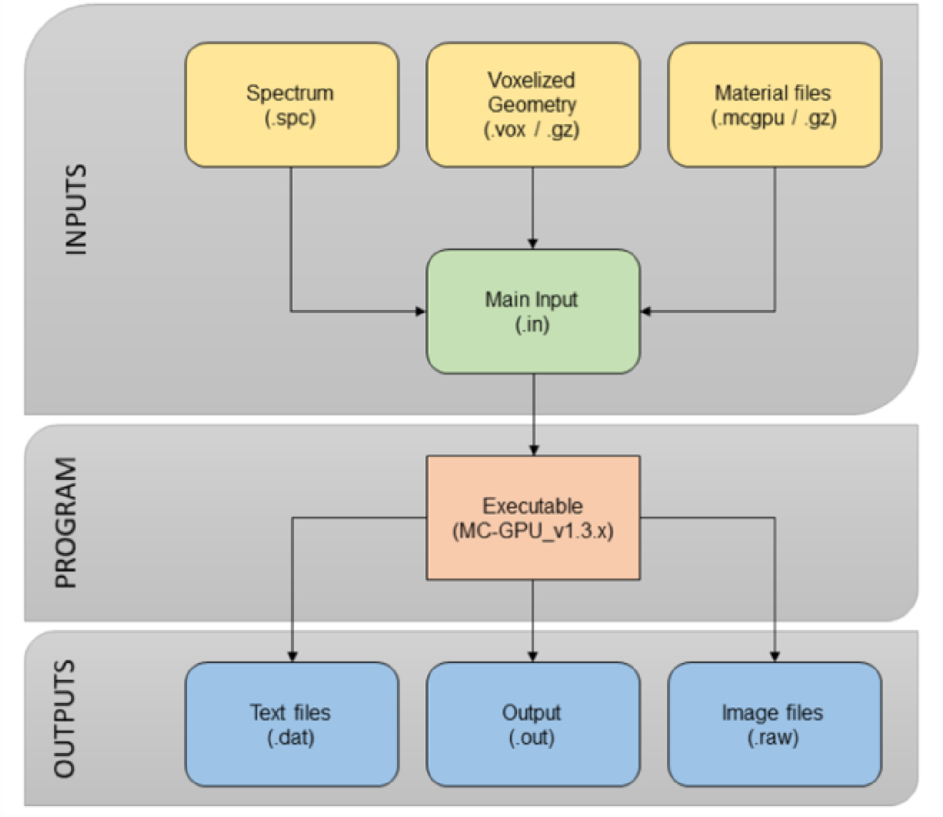


Figure 4: MC-GPU work-flow outline [57]

4.2.1 Input block

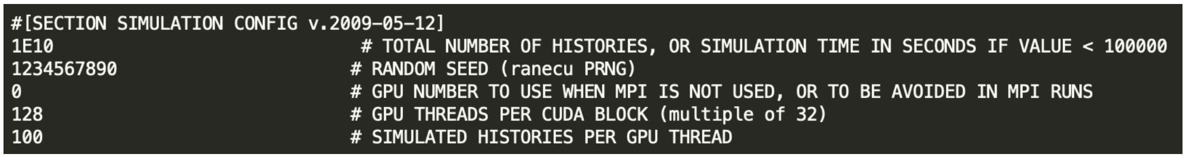
In the input block of the MC-GPU program, all the parameters required for the simulation are specified through a simulation input file. This file is manually created by the user and contains information that describes various aspects of the simulation setup. The input file is structured into different sections, each containing specific sets of parameters. Let's explore the different sections present in the simulation input file.

4.2.1.1 Section simulation configuration

In this section, the user defines the values of various parameters related to the precision, accuracy, and time performance of the simulations. These parameters are presented below in Table 2, and a screenshot of the simulation configuration section in the input file is provided in Figure 5.

Table 2: Simulaction configuration section

Total number of histories (N) or simulation time in s: Total number of photons to be simulated. If N < 1E5 the program interprets the value as the simulation time in seconds.
Random seed: This number is the seed of the random generation algorithm used by MC-GPU.
GPU number to use when MPI is not used, or to be avoided in MPI runs: MPI stands for Message Passing Interface, which is a communication protocol used for parallel programming. In this case, in utilizing multiple GPUs. Each GPU is assigned a unique ID (scalar number) for identification purposes. The interpretation of this parameter depends on the number of GPUs employed in the simulation. When only one GPU is used, the ID indicates the specific GPU on which the simulation should be executed. However, when MPI is employed to distribute the simulation across multiple GPUs, the ID indicates which GPUs should not be excluded from the computation. This can be useful in cases where certain GPUs are slower or less efficient, and the user wishes to avoid their usage in the simulation.
GPU threads per CUDA block and simulated histories per GPU thread: The performance and computation time of GPU processors are influenced by various parameters related to their internal architecture. These parameters determine how efficiently the GPU can process and execute tasks. They are explained in section 4.1.2.



```
#[SECTION SIMULATION CONFIG v.2009-05-12]
1E10                             # TOTAL NUMBER OF HISTORIES, OR SIMULATION TIME IN SECONDS IF VALUE < 100000
1234567890                      # RANDOM SEED (ranecu PRNG)
0                               # GPU NUMBER TO USE WHEN MPI IS NOT USED, OR TO BE AVOIDED IN MPI RUNS
128                             # GPU THREADS PER CUDA BLOCK (multiple of 32)
100                             # SIMULATED HISTORIES PER GPU THREAD
```

Figure 5: Screenshot of the section simulation configuration in the input file.

4.2.1.2 Section source

In this section, the user can select the spectra of the X-ray beam used for the simulation and define a set of parameters related to the position of the beam source relative to the patient, as well as its shape and direction. The parameters are presented in Table 3, and a screenshot of the beam source section in the input file is depicted in Figure 6.

Table 3: Source section

X-ray energy spectrum file:

The user needs to introduce the path to the file containing the X-ray spectra to be simulated. The X-ray spectra have to be generated outside MC-GPU. MC-GPU only supports reading a file with the specific format and extension spc. We hereafter refer to this file as spectrum.spc. In section 4.4 we explain how is the X-ray spectra file generated.

Source position:

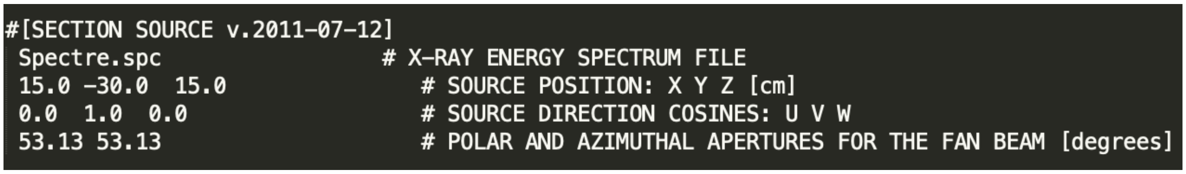
The relative position of the beam source to the patient is determined by the user by a set of coordinates. Namely, x_{source}, y_{source} and z_{source} in units of cm. These coordinates are the cartesian coordinates of the internal coordinate system of the program. In section 4.7 we explain how is the system of coordinates defined.

Source direction cosines:

The beam source can be pointing in any direction of the 3D space defined by the system of coordinates. The direction is defined by a normal vector and its values need to be introduced by the user.

Beam angular apertures:

The shape of the X-ray beam field in MC-GPU is computationally collimated to a pyramidal shape to produce a rectangular field on the detector plane. The shape is defined by its aperture angles, which are called polar and azimuthal as shown in Figure 7. The user can introduce its values in units of degrees. If the input is negative the program automatically chooses the angular apertures to cover the whole detector.



```
#[SECTION SOURCE v.2011-07-12]
 Spectre.spc                # X-RAY ENERGY SPECTRUM FILE
 15.0 -30.0  15.0              # SOURCE POSITION: X Y Z [cm]
 0.0  1.0  0.0                 # SOURCE DIRECTION COSINES: U V W
 53.13 53.13                   # POLAR AND AZIMUTHAL APERTURES FOR THE FAN BEAM [degrees]
```

Figure 6: Screenshot of the section source in the input file.

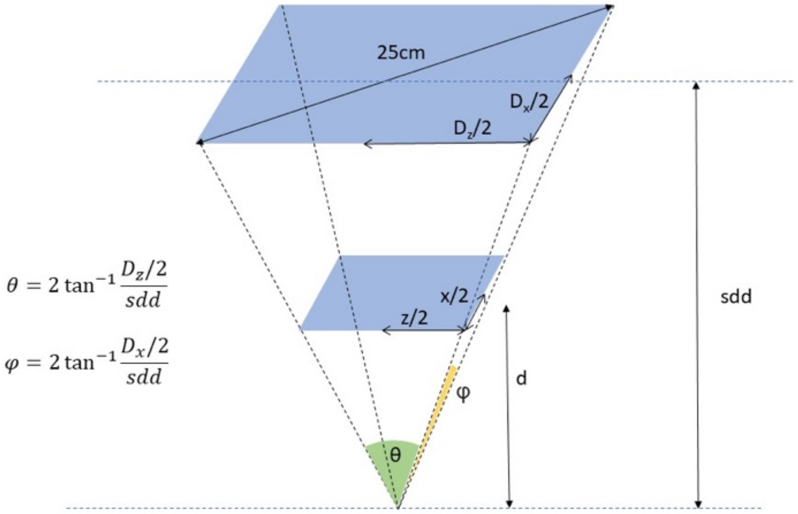


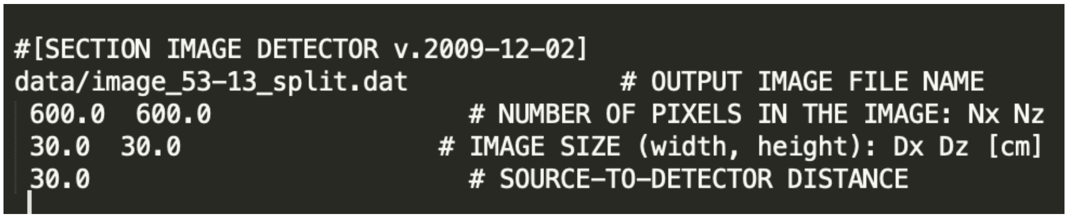
Figure 7: Sketch of a pyramidal beam source

4.2.1.3 Image detector section

MC-GPU generates synthetic radiographic images using a detector that simulates the image detector found in real IR (image receptor) equipment. The detector plane is positioned and fixed at a specified distance directly in front of the source focal spot, with the collimated cone beam directed towards the geometric center of the detector. The parameters for this setup are presented in Table 4, and a screenshot of the image detector section in the input file is displayed in Figure 8.

Table 4: Image detector section

Output image file name:	Name of the file in which the image will be saved in raw format.
Number of pixels in the detector image:	Number of pixels in the x and z direction.
Detector image size:	Width (D_x) (x-axis) and height (D_z) (z-axis) in units of cm of the detector image.
Source-to-detector distance:	Perpendicular distance from the point beam source to the detector in cm.



```
#[SECTION IMAGE DETECTOR v.2009-12-02]
data/image_53-13_split.dat             # OUTPUT IMAGE FILE NAME
 600.0  600.0                # NUMBER OF PIXELS IN THE IMAGE: Nx Nz
 30.0  30.0                # IMAGE SIZE (width, height): Dx Dz [cm]
 30.0                        # SOURCE-TO-DETECTOR DISTANCE
```

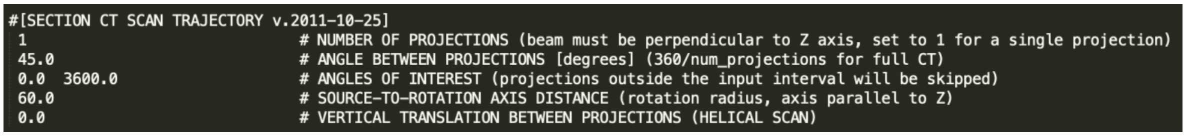
Figure 8: Screenshot of the image detector section.

4.2.1.4 CT scan trajectory section

MC-GPU provides the option to simulate either a single projection image or a complete Computed Tomography (CT) scan. The relevant parameters for these simulations are presented in Table 12, and a screenshot of the CT scan trajectory section in the input file is displayed in Figure 9.

Table 5: CT scan trajectory section

Number of projections:	Number of projections to be simulated.
Angle between projections:	Angles between projections in degrees
Angles of interest:	Interval of angles of interest. Projections outside the input interval will be skipped.
Source-to-rotation axis distance:	Distance between the source and the axis of rotation.
Vertical translation between projections:	Translation of the patient in the vertical direction between projections in cm.



```
#[SECTION CT SCAN TRAJECTORY v.2011-10-25]
 1                              # NUMBER OF PROJECTIONS (beam must be perpendicular to Z axis, set to 1 for a single projection)
 45.0                           # ANGLE BETWEEN PROJECTIONS [degrees] (360/num_projections for full CT)
 0.0  3600.0                    # ANGLES OF INTEREST (projections outside the input interval will be skipped)
 60.0                           # SOURCE-TO-ROTATION AXIS DISTANCE (rotation radius, axis parallel to Z)
 0.0                            # VERTICAL TRANSLATION BETWEEN PROJECTIONS (HELICAL SCAN)
```

Figure 9: Screenshot of the CT scan trajectory section.

As we will discuss later, an IR (Image Reconstruction) procedure typically consists of a significant number of single projections. Each individual projection may have different X-ray spectra or beam source angles. However, it is important to note that, as indicated in Table 12 and Figure 9, the X-ray spectra cannot be modified between different CT projections. Consequently, simulating an entire IR procedure solely using a single input file with the CT scan trajectory section is not feasible in MC-GPU. To address this issue, a new version of MC-GPU has been developed, which will be introduced in detail in 4.3.1.

For simulation purposes, the parameter values in this section remain fixed and are selected to perform a single projection in each simulation, but it is worth mentioning that MC-GPU is prepared to simulate a full CT scan. These fixed parameter values are illustrated in Figure 9.

4.2.1.5 Dose deposition section

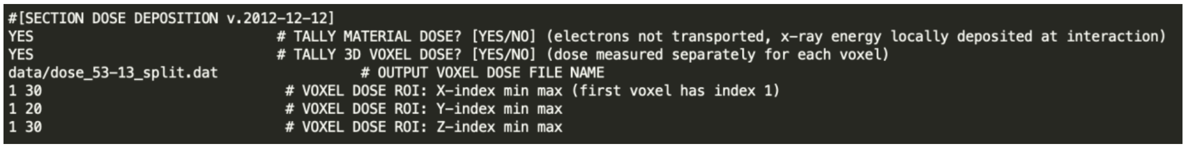
MC-GPU offers two tally options: an image tally of the X-ray projections and the radiation dose deposited within the patient model.

In the image tally, the formation of the image is based on the accumulation of total energy entering a user-defined 2D grid of pixels. The pixel values are expressed in units of eV/cm^2. At the end of the simulation, four different images are generated, representing the signal produced by non-scattered X-rays (those that did not interact between the source and the detector), X-rays that experienced a single Compton (inelastic) interaction, X-rays that underwent a single Rayleigh (elastic) interaction, and multi-scattered X-rays.

For the radiation dose tally, users have the option to calculate the radiation dose deposited within each material or within a user-defined volumetric region-of-interest (ROI). The dose is calculated by tallying the total deposited energy for each X-ray track simulation within each voxel of the geometry. The corresponding parameters for these tally options are presented in Table 6, and a screenshot of the dose deposition section in the input file is displayed in Figure 10.

Table 6: Dose deposition section

Tally material dose:
The options are YES or NO.
Tally 3D voxel dose:
The options are YES or NO.
Output voxel dose file name:
Name of the output file in which the dose in the region of interest will be saved.
Voxel Dose ROI: X-index, Y-index and Z-index:
User-defined volumetric region-of-interest (ROI) is determined by the first and last voxels in each of the axis of the coordinate system.



```
#[SECTION DOSE DEPOSITION v.2012-12-12]
YES                             # TALLY MATERIAL DOSE? [YES/NO] (electrons not transported, x-ray energy locally deposited at interaction)
YES                             # TALLY 3D VOXEL DOSE? [YES/NO] (dose measured separately for each voxel)
data/dose_53-13_split.dat                # OUTPUT VOXEL DOSE FILE NAME
1 30                             # VOXEL DOSE ROI: X-index min max (first voxel has index 1)
1 20                             # VOXEL DOSE ROI: Y-index min max
1 30                             # VOXEL DOSE ROI: Z-index min max
```

Figure 10: Screenshot of the dose deposited section.

4.2.1.6 Voxelized geometry section

In this section, the user provides the path to the voxelized geometry file for the simulation. The input for this path is specified by the user, as depicted in the screenshot of the voxelized geometry section in the input file (Figure 11).

```
#[SECTION VOXELIZED GEOMETRY FILE v.2009-11-30]
18oct_split.vox                     # VOXEL GEOMETRY FILE (penEasy 2008 format; .gz accepted)
```

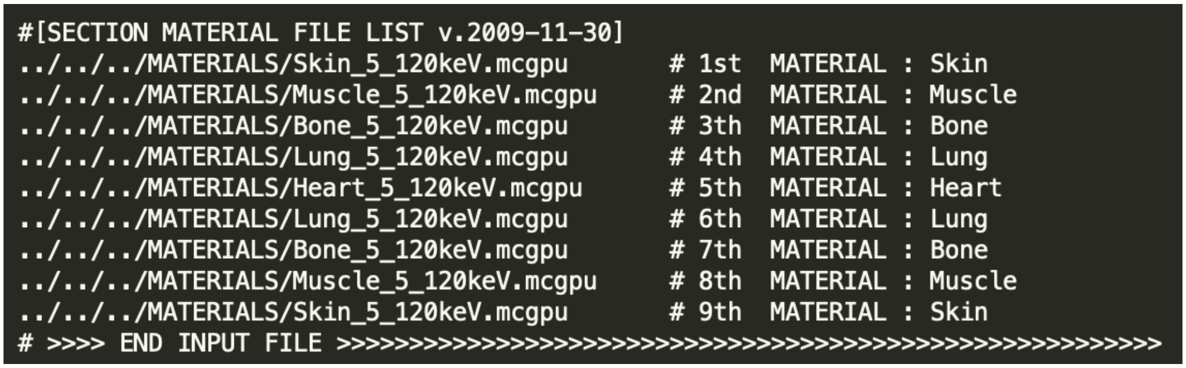
Figure 11: Screenshot of the voxelized geometry section.

4.2.1.7 Material files list section

In this section, the user specifies the path to the various materials required for the simulation. These material files contain the necessary information for the physics to be simulated. The cross-section values for each possible interaction are provided for different energy ranges.

MC-GPU uses a database of material properties based on the data from PENELOPE. To use the material files from PENELOPE 2006 with MC-GPU, a subroutine auxiliary utility included in the MC-GPU package can be used to convert them into MC-GPU material files. In case new materials need to be generated, the PENELOPE subroutine for generating new materials can be employed and subsequently converted into MC-GPU material files.

Figure 12 displays a screenshot of the material files list section in the input file, providing an example of how it is structured.



```
#[SECTION MATERIAL FILE LIST v.2009-11-30]
../../../MATERIALS/Skin_5_120keV.mcgpu       # 1st  MATERIAL : Skin
../../../MATERIALS/Muscle_5_120keV.mcgpu     # 2nd  MATERIAL : Muscle
../../../MATERIALS/Bone_5_120keV.mcgpu       # 3th  MATERIAL : Bone
../../../MATERIALS/Lung_5_120keV.mcgpu       # 4th  MATERIAL : Lung
../../../MATERIALS/Heart_5_120keV.mcgpu      # 5th  MATERIAL : Heart
../../../MATERIALS/Lung_5_120keV.mcgpu       # 6th  MATERIAL : Lung
../../../MATERIALS/Bone_5_120keV.mcgpu       # 7th  MATERIAL : Bone
../../../MATERIALS/Muscle_5_120keV.mcgpu     # 8th  MATERIAL : Muscle
../../../MATERIALS/Skin_5_120keV.mcgpu       # 9th  MATERIAL : Skin
# >>>> END INPUT FILE >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```

Figure 12: Screenshot of the material files list section.

4.2.2 Program block

To run MC-GPU, you will need the CUDA libraries and the GNU GCC compiler. MC-GPU uses CUDA to leverage the processing power of NVIDIA GPUs. The computations are implemented in standard C. The X-ray transport simulation can be performed either in parallel on an NVIDIA GPU using CUDA or sequentially executed on a CPU. Additionally, if available, multiple GPUs can be used using MPI (Message Passing Interface).

Before running the code, it needs to be compiled. This can be easily accomplished with a script provided in the MC-GPU package. Once compiled, the code can be executed in a command shell.

4.2.3 Output block

Four output files are generated by MC-GPU when the simulation is finished: the standard output file, the voxel dose ROI tally report, the material total dose tally report and the image detector files.

4.2.3.1 Standard output file

The main output file is a text file that encompasses all the information from the simulation. Firstly, it provides details about the configuration and execution of the simulation on either GPUs or CPUs.

Additionally, it reports the parameter values specified in the simulation input file. Furthermore, the output file includes a table presenting the average dose per material. The table contains the following information for each material: material ID, dose (eV/g·hist), twice the standard deviation, relative error (%), energy deposited (eV/hist), mass (g), and the number of voxels.

Additionally, the total dose within the defined regions of interest (ROIs) is provided. Furthermore, information about the voxel with the maximum dose within the ROI is reported. This includes the voxel dose with twice the standard deviation in units of eV/g/hist, the energy deposited in the voxel (eV/hist), the density (g/cm^3), the mass (g), and the voxel coordinates relative to the reference system.

4.2.3.2 Voxel dose ROI tally report

The dose per voxel within the ROI is reported in a raw binary format, represented as 32-bit float values. Additionally, a separate raw file is provided, which contains twice the standard deviation per voxel.

4.2.3.3 Material total dose tally report

The total averaged dose per material is reported as an ASCII text file in a table format, containing the same values mentioned in section 4.2.3.1 of the standard output file. However, the difference is that this table does not include the row and column name entries present in the standard output file. This formatting choice is made to enhance readability and facilitate post-processing tasks.

4.2.3.4 Image detector files

At the end of the simulation, the code generates several output files. The tallied 3D dose distribution and the final simulated images are stored in raw binary format as 32-bit float values. The image data consists of five consecutive images, including the total image (scatter+primaries), primary particles, Compton, Rayleigh, and multi-scatter images.

In addition, the program generates ASCII text files containing the simulated images and the dose at the Z plane where the X-ray source is located. These ASCII output files can be easily visualized using the GNUPLOT [59] scripts provided with MC-GPU. The header section at the beginning of these text files includes the necessary information to facilitate the reading of the raw binary files with programs such as IMAGEJ [60], OCTAVE [61], or others.

The pixel and voxel dose data values are stored in a manner where the x coordinate increases first, followed by the y coordinate, and finally the z coordinate.

To obtain organ doses, post-processing of the output dose file is necessary, with knowledge of the voxel-to-organ mapping to determine which voxel corresponds to each organ.

4.3 PyMCGPU-IR

As explained in Part II, the main objective of this work is to develop a software tool based on the Monte Carlo program MC-GPU for assessing the skin dose in patients undergoing interventional radiology

(IR) procedures. Achieving this objective involves the development and validation of PyMCGPU-IR, a skin dose calculation tool for IR procedures based on MC-GPU.

During the development of PyMCGPU-IR, several modifications have been made to MC-GPU, along with the creation of program subroutines to automate all the necessary steps for calculating the skin dose. These modifications will be presented in two main steps. Firstly, the development of MCGPU-IR. Secondly, the development of PyMCGPU-IR, and extension of MCGPU-IR.

First, let's explain the main processes that need to be automated for the skin dose calculation, and then we will introduce both MCGPU-IR and PyMCGPU-IR, presented in sections 4.3.1 and 4.3.2, respectively.

From a general perspective, the primary requirement for a Skin Dose Calculation (SDC) tool is to automate the necessary steps for dose calculation. The processes that need to be automated in MC-GPU are as follows:

a) Input file generation:

   The input files for MC-GPU must be generated manually by inputting the parameter values mentioned in section 4.2.1. However, simulating a complete IR procedure can be a cumbersome task, especially when there are numerous events to simulate, each requiring the generation of individual input files. Consequently, automation of input file generation becomes essential.

b) Spectrum file generation:

   Each event may have its own X-ray spectrum, as the kV and filters are automatically adjusted during the intervention using the automatic control exposure system (AEC). In MC-GPU, the X-ray spectrum is provided through an input file, so the generation of the X-ray spectrum must be done externally. Therefore, a subroutine is required in MC-GPU to automatically generate the X-ray spectrum. Section 4.4 explains the different approaches used in this work to achieve this.

c) Calculation of the dose conversion factor:

   MC-GPU calculates the dose as deposited energy per simulated history (eV/(g·history)). However, to obtain the absorbed dose in absolute values (Gy), a normalization factor must be applied, as detailed in section 4.6. This factor is determined through a second simulation with MC-GPU, meaning that two simulations are required to obtain the dose in Gy units for a single event. A subroutine that automatically prepares and executes both simulations was written to address this issue

d) MC-GPU execution:

Once the input files are generated, they need to be executed one by one. Automating this process allows for the sequential execution of all the generated input files with just one execution instruction.

e) Dose retrieval from the outputs:

Upon finalization of the simulation, the doses must be retrieved and calculated from the MC-GPU outputs. Automation of this process is essential.

4.3.1 MCGPU-IR

In 2013, MC-GPU beta [62] was developed as an extension of the original MC-GPU. MC-GPU beta incorporated the calculation of staff doses resulting from patient scattered radiation. It also introduced the capability to automatically simulate multiple irradiation events, providing both total organ doses and individual results for each event. Beam parameters such as kV, added filtration, beam position, and angulation were taken into account for each irradiation event.

As a further development of MC-GPU beta, MCGPU-IR was improved within the framework of the European projects MEDIRAD and PODIUM [41, 63]. The following major improvements were implemented in MCGPU-IR:

- MCGPU-IR includes a set of functions to automatically determine optimal values for GPU calculations, such as the number of blocks per kernel, number of threads per block, and number of histories per thread to be simulated on the GPU. Please refer to section 4.1.2 for more details.
- MC-GPU provides doses as deposited energy per simulated history (eV/(g·history)). In MCGPU-IR, a normalization factor is introduced to convert deposited energy to absorbed dose in absolute values (Gy). See section 4.6 for further information.
- MC-GPU does not account for the rotation of the C-arm, which is a limitation when simulating IR procedures, as the C-arm is typically rotated during the intervention. In MCGPU-IR, two new parameters are introduced to account for the rotation of the source with respect to the patient in two different planes of rotation. Please see section 4.7 for details.

The aforementioned features 2 and 3 in MCGPU-IR are introduced in a separate file called the trigger file, which complements the input file. The parameters included in the trigger file are presented in Table 7, and a screenshot of the trigger file is shown in Figure 13.

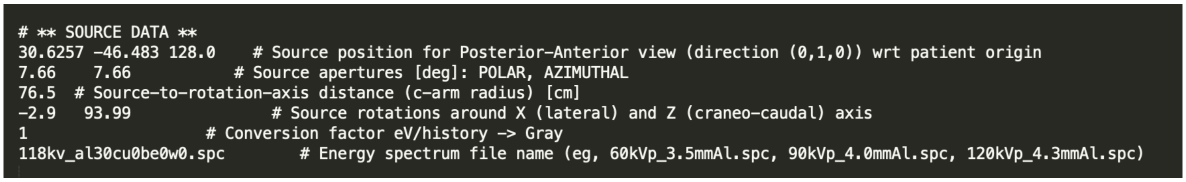
Table 7: Parameters of the trigger file.

Source position:	Same as in 4.2.1.2.
Beam angular apertures:	Same as in 4.2.1.2.
Source-to-rotation-axis distance:	Distance from the source to the center of rotation of the source. This point is usually the isocenter and is given in the RDSR..
Source rotations around x (lateral) and z (craneo-caudal) axis:	Angular rotations in units of degrees of the beam source around the x-axis and z-axis, see section 4.7.
Conversion factor:	Normalization factor to convert dose values per simulated history to absolute values, see section 4.6.
X-ray energy spectrum file:	Same as in 4.2.1.2.

Readers may observe that some of the parameters mentioned above are already included in various sections of the input file presented in section 4.2.1. Consequently, certain parameters need to be duplicated in both the input file and the trigger file. In cases where different values are specified for the same parameter in the input and trigger files, MCGPU-IR will prioritize the values from the trigger file. This is particularly important to consider for the beam angular apertures. In MC-GPU, if the beam angular aperture values are negative, the beam shape is automatically adjusted to cover the entire detector. However, in MCGPU-IR, the beam angular apertures cannot have negative values. Hence, the beam angular apertures must always be calculated and entered in the trigger file.

4.3.2 PyMCGPU-IR

As a further step forward, we developed PyMCGPU-IR, a Python application for MCGPU-IR that includes patient dose calculations in interventional radiology. It can read several types of input data files (e.g. RDSR) to retrieve the required information about each irradiation event and transform them



```
# ** SOURCE DATA **
30.6257 -46.483 128.0    # Source position for Posterior-Anterior view (direction (0,1,0)) wrt patient origin
7.66    7.66           # Source apertures [deg]: POLAR, AZIMUTHAL
76.5  # Source-to-rotation-axis distance (c-arm radius) [cm]
-2.9   93.99               # Source rotations around X (lateral) and Z (craneo-caudal) axis
1                   # Conversion factor eV/history -> Gray
118kv_al30cu0be0w0.spc        # Energy spectrum file name (eg, 60kVp_3.5mmAl.spc, 90kVp_4.0mmAl.spc, 120kVp_4.3mmAl.spc)
```

Figure 13: Screenshot of the trigger file.

into MCGPU-IR input and trigger files. The application then proceeds to launch the simulation, and finally retrieves the radiation doses from its output. As MCGPU-IR does not parse RDSR files, before PyMCGPU-IR all these processes had to be done manually. A schematic diagram of the PyMCGPU-IR workflow is shown in Figure 14.

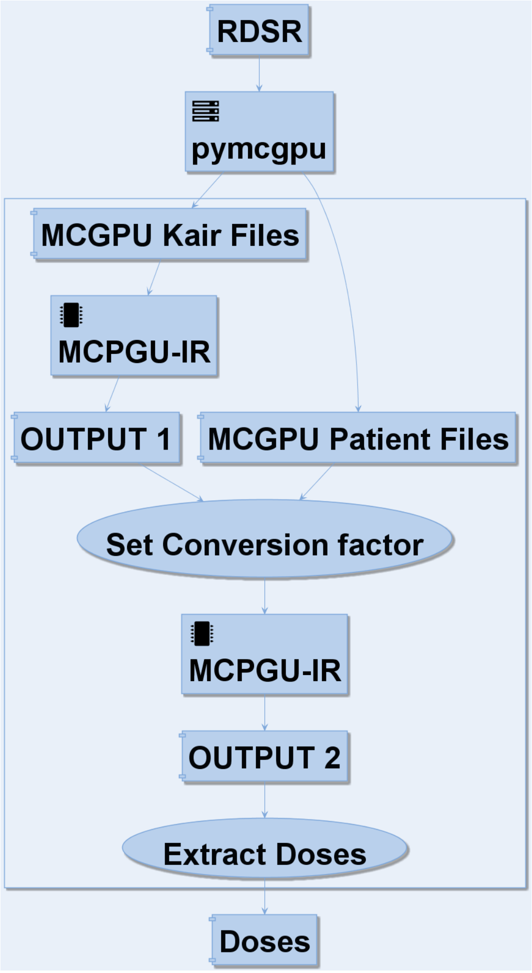


Figure 14: PyMCGPU-IR workflow

4.3.3 PyMCGPU-IR features

4.3.3.1 Supported RDSR file formats

The Radiation Dose Structured Report (RDSR) is used to store information regarding the dose of each irradiation event in an interventional procedure, including patient information, procedure details, and manufacturer-specific data. The RDSR format is part of the DICOM® standard, and each machine manufacturer provides a unique DICOM® conformance statement specific to their machine models or software versions. These conformance statements are usually publicly available on the manufacturers' websites, such as Philips DICOM® or Siemens DICOM®.

The generation of RDSR is performed by the X-Ray machine and is stored internally or within the hospital's Picture Archiving and Communication System (PACS). The specific storage location of the RDSR depends on the hospital's infrastructure and setup, based on how the RDSR will be retrieved later. In some cases, the setup may not allow for retrieval at all.

Although the RDSR is generated in a standardized format, its retrieval may not always be in the same format. The format may be determined by the PACS or the user's access privileges. If retrieval is possible, some PACS systems will provide the original DICOM® format, while others may offer access to the RDSR in a converted markup representation (XML). Additionally, there are instances where the RDSR can only be obtained as an exported spreadsheet.

To address these variations, PyMCGPU-IR supports reading RDSR in different formats. This includes the standard DICOM® format as well as spreadsheet formats. Furthermore, the spreadsheet format in PyMCGPU-IR allows users the option to easily create their own reports.

In summary, PyMCGPU-IR supports the following file formats and extensions: DICOM (.dcm, .sr), spreadsheets stored in the Microsoft Excel file format (.xls), and spreadsheets stored in the Office Open XML file format (.xlsx).

4.3.3.2 Information retrieval from RDSR

PyMCGPU-IR is designed to extract all the necessary information for the dose calculation from the Radiation Dose Structured Report (RDSR). Figure 15 illustrates the specific information that is extracted from the RDSR.

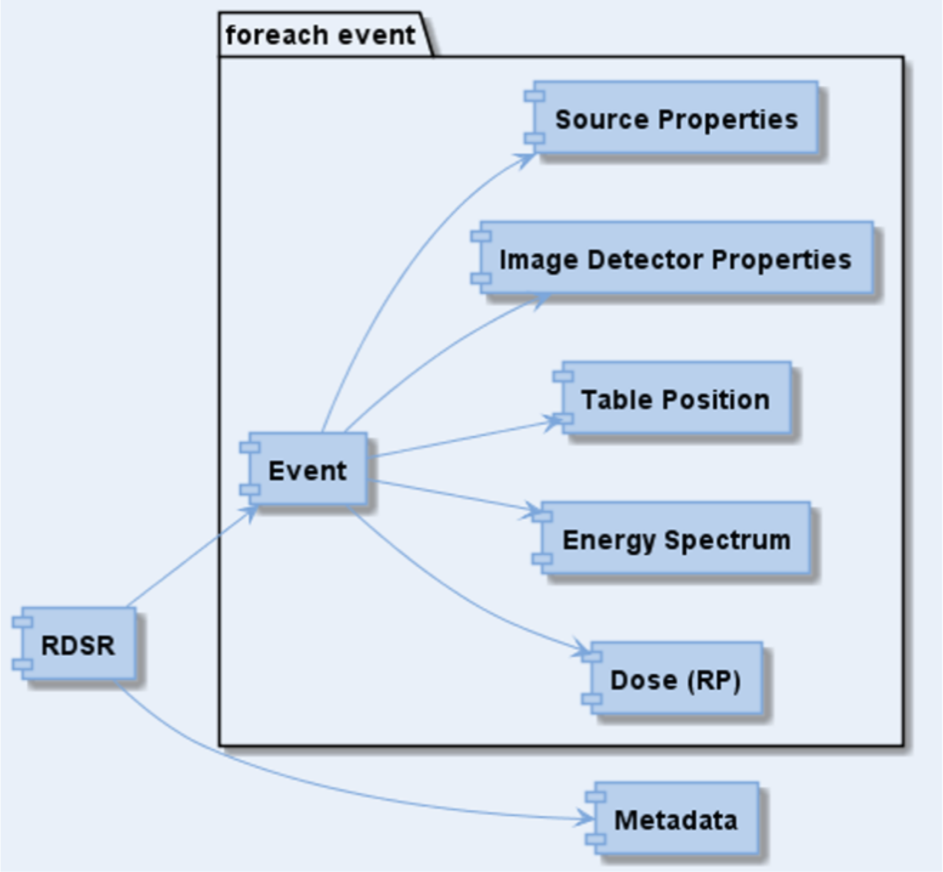


Figure 15: Information extracted by PyMCGPU-IR from the RDSR

Once the information is retrieved, it needs to be converted into the appropriate representation or file format required by the dose calculation program, MCGPU-IR. Specifically, it needs to be transformed into the input and trigger files as described in sections 4.2.1 and 4.3.1. For a more detailed explanation of this conversion process, please refer to section 4.7.

4.3.3.3 Unification of RDSR divergences

Even though RDSR is a standardized format, it is possible for different machines to use various approaches to represent the values of specific fields. The RDSR may also contain machine-specific details or deviations from the standard. In cases where machine-specific information is available, it provides a more accurate representation of the event. For instance, the distance from the source to the detector might be stored as "Final Distance" in some machines, instead of the standard label "Distance to Detector." Another example pertains to the representation of the reference point position, which can vary in naming across different machines. For instance, it may be referred to as "[distance]cm from Isocenter towards source" in some cases and "[distance]cm below BeamIsocenter" in others.

Additionally, the position of shutters is not always reported, even though it offers a more accurate representation, and both shutters and "Dose Area Product" are defined in the DICOM® standard.

PyMCGPU-IR addresses these variations by supporting different representations, mapping non-standard fields to their standard equivalents, or utilizing specific machine details whenever possible. The diversity of machines and models available will require further developments in this process. The objective is to enable advanced users or developers to easily customize PyMCGPU-IR to accommodate specific situations, as we have done in our work. It is not anticipated that a universal system will be established in the near future, but rather a flexible system that can be adapted to different scenarios.

4.3.3.4 RDSR Complementation

In some instances, the RDSR may lack certain required information. This can occur due to anonymization, resulting in missing patient gender, or it may be incomplete with missing patient height or weight if not entered by the clinician. Additionally, certain details like the inherent filtration of the X-ray beam source may not be included in the RDSR, even though it is necessary for accurate dose calculations. To overcome these challenges, PyMCGPU-IR provides support for reading an additional file in JavaScript Object Notation (JSON) format created by an "advanced user". This file serves to supplement the information obtained from the RDSR. It can include details regarding the patient, room geometry, and/or X-ray source, ensuring a more comprehensive dataset for dose calculations.

4.3.3.5 Clustering of events

In a typical interventional radiology (IR) procedure, once the X-ray beam is positioned on the region of interest and adjusted to achieve optimal image quality, it often remains in the same configuration for a period of time, performing single irradiation events with only minor modifications. This pattern continues until irradiations from a different direction or a new region of interest are required. As a result, there can be sets of events with slight variations in radiological parameter values. MCGPU-IR calculates doses for each individual event. To reduce computation time, a clustering algorithm has been developed to group these sets of events with negligible differences, effectively reducing the total number of events to be simulated by MCGPU-IR.

The proposed algorithm groups events that share similar characteristics based on specific criteria. A trigger value is assigned to each parameter, representing the maximum allowable difference between two events to be grouped together. The algorithm compares the parameter values of two events and checks if the differences are within the trigger value thresholds. If all differences are below or equal to the corresponding trigger values, the events are grouped together. The grouping process is performed consecutively. When two events are grouped, a group-event representing the group is generated, and its parameter values are set as the mean parameter values of both events. The next event is then compared to the existing group-events, and if it satisfies the grouping criteria, the group-event is updated accordingly. Table 8 provides the trigger values defined for each parameter.

It's important to note that the trigger levels in our clustering algorithm are not intended to find equivalent spectra. The primary objective is to group events with "same" or "almost the same" (within the trigger levels) radiological parameters in order to reduce calculation time by simulating fewer events. We evaluate whether one event is similar or dissimilar to a previous event based on these parameters and their corresponding intervals, as shown in Table 8, which reflect real-world practice.

Table 8: Trigger values of the clustering algorithm

Source Beam Angulations (º)		Source Beam Filtrations (m)					Source Beam Position (mm)			Source Beam Apertures (º)	
Primary Angle	Secondary Angle	Energy (kVp)	Al	Cu	Be	W	x	y	z	Polar	Azimuthal
0.2	0.2	0.5	0.2	0.2	0.2	0.2	0.5	0.5	0.5	0.5	0.5

4.3.3.6 Reporting patient doses

The main objective of the simulation is to calculate patient doses, which are reported for individual organs or with higher granularity for graphical visualization. The granularity is determined by the voxelized geometry used to represent the patient. The voxel size defines the spatial resolution of the organ volumes and enables their identification. MCGPU-IR provides output for accumulated doses per tag and event, as well as total accumulated doses, and stores the deposited voxel doses per irradiation event as a file.

The simulation uses geometries with 27 tags or materials (refer to section 4.1.4) corresponding to different body parts such as organs, bones or watter. The accumulated dose can be retrieved from the output of MCGPU-IR. Tags are represented using numerical values. The output is in a tabular format once it is ready, representing the accumulated dose. PyMCGPU-IR parses the output to extract the dose values for each tag and event. It then stores the aggregated doses in a single file, identifying each event by its date and time. This file can be saved as a spreadsheet (Excel) or in JSON format.

4.3.3.7 Supported voxelized phantoms

PyMCGPU-IR uses the Rex and Regina phantoms, which are described in section 4.1.4, to model the patient. In PyMCGPU-IR, the original 145 segmented objects in the phantoms are simplified and grouped into 27 different tags or materials.

The Rex phantom has voxel dimensions of 0.21 x 0.21 x 0.80 cm^3, resulting in a voxel volume of 36.53 mm^3. The Regina phantom, on the other hand, has voxel dimensions of 0.18 x 0.18 x 0.48 cm^3, resulting in a voxel volume of 15.25 mm^3.

For each material, PyMCGPU-IR allows the calculation of both the maximum and mean dose, as well as the location of the maximum dose within the patient. Additionally, both phantoms can be rescaled based on the patient's height and weight to achieve a better fit to the patient's size.

4.4 X-ray spectra generation

The radiation beams used in Interventional Radiology (IR) procedures are produced by an X-ray tube, which consists of a cathode, an anode, and a power supply, as illustrated in Figure 16. The cathode is responsible for generating free electrons through thermionic emission, while the anode serves as the target where the emitted electrons interact to generate X-rays. The cathode and anode are enclosed within a glass, ceramic, or metal-ceramic envelope, creating a vacuum environment. The tube housing effectively contains the X-rays, preventing their emission into the surrounding room.

The cathode, functioning as the negative electrode, is composed of a tungsten filament. When a high current passes through the filament, it becomes extremely hot, resulting in the release of a cloud of free electrons, known as a space charge. This process, known as thermionic emission, directly correlates the number of generated free electrons with the intensity of the current flowing through the filament. Consequently, a higher current leads to a greater number of free electrons being generated. On the other hand, the anode, serving as the positive electrode, consists of a thick Copper rod with a small piece of tungsten target located at its end (see Figure 16).

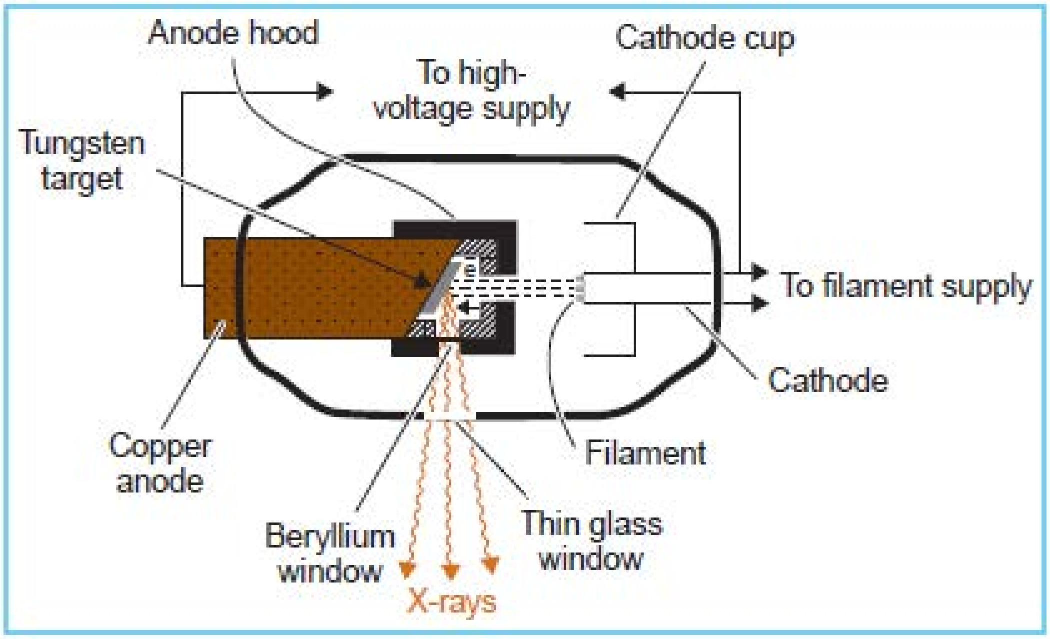


Figure 16: Components of an X-ray tube `https://radiologykey.com/production-of-x-rays-5/`

When a high voltage is applied between the cathode and anode in the X-ray tube, the free electrons generated in the cathode are accelerated towards the anode, attaining high velocities and energies. The velocity and energy of the electrons before they strike the anode target depend on the applied voltage. Higher voltage results in higher electron energy and velocity. Upon collision with the anode, the electrons are converted into heat, with only a small portion producing X-rays. The X-rays are generated through Bremsstrahlung radiation, which occurs when the accelerated electrons are deflected by the attractive force of the tungsten nucleus.

The anode angle refers to the angle between the vertical axis and the target surface of the anode. Most X-ray tubes have an anode angle of 6-20°. This angle affects the size of the projected focal spot and the distribution of X-ray intensity within the beam. The focal spot is the area on the anode surface where the electron beam from the cathode is directed. The anode angle controls the size of the projected focal spot, with a smaller spot resulting in sharper image details.

Filtration is placed between the X-ray output of the anode and the patient. There are two types of filtration: inherent filtration and added filtration. Inherent filtration is provided by components within the X-ray tube, such as the window, housing, and cooling oil. Added filtration involves the use of metal sheets to modify the X-ray spectrum. Filtration can be adjusted to eliminate low-energy photons that could harm the patient's skin and do not provide useful information to the diagnostic image.

Collimators are used to shape the radiation beam and limit its field size. They help to control the size and shape of the beam emerging from the X-ray machine. In MC-GPU, the beam shape is pyramidal.

To generate the beam, the parameters needed include the tube voltage, anode angle, and information about the inherent and added filtration. As explained in section 4.3, the X-ray spectra in MC-GPU need to be generated externally and provided as an input file.

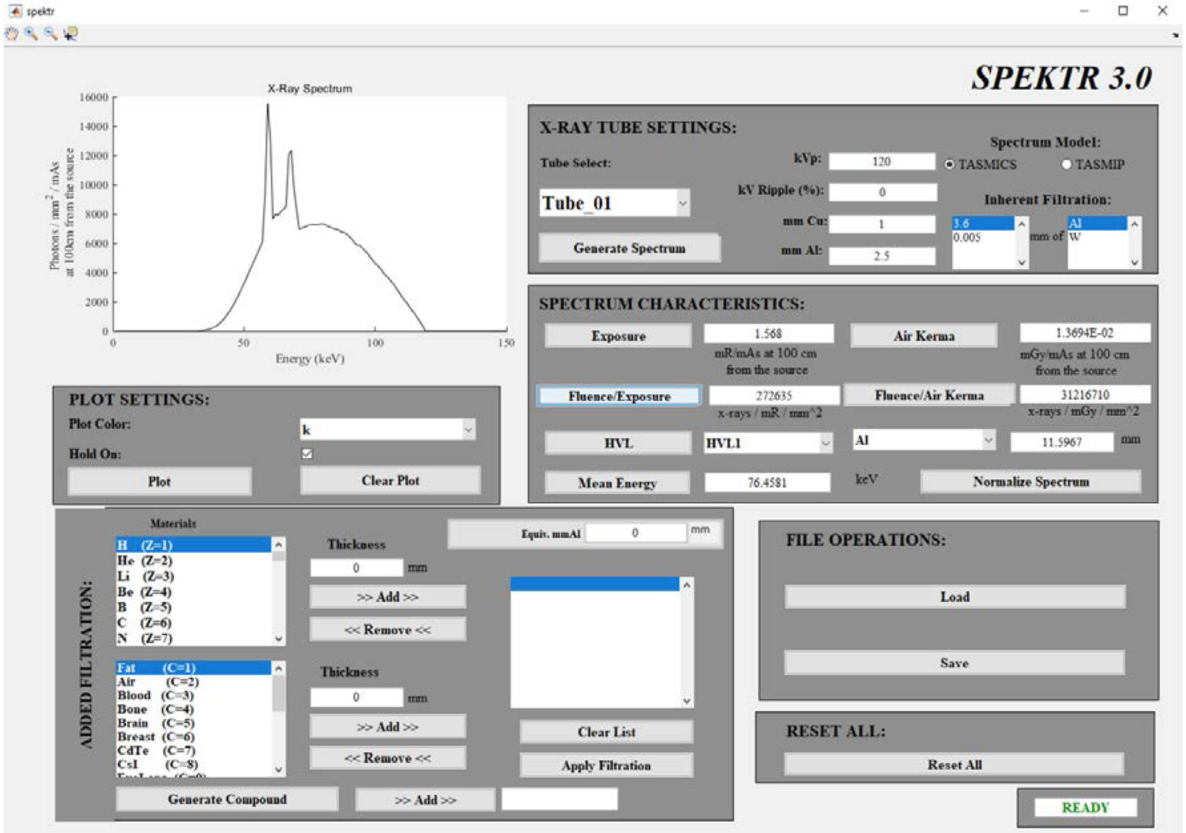
In this work, we employed two different approaches to generate the X-ray spectrum file. Specifically, we used the XComp5r software [64] for the validation of MC-GPU, and we developed a custom script based on the Spektr 3.0 [65] software to fulfill the basic requirement of spectrum file generation, as presented in section 4.3.

The XComp5r software is designed for calculating diagnostic X-ray spectra for DC voltages ranging from 20 to 150 kV and anode angles from 0 to 45º. It provides the option to model filtration using eight different materials with various thicknesses, including Be, Al, Cu, Sn, Pb, water, PMMA, and transformer oil. The software calculates X-ray bremsstrahlung spectra and generates an output file in ASCII format with the X-ray spectrum energy distribution.

However, the output file from XComp5r requires modifications to be used as input for MC-GPU. Specifically, the file needs to contain only the radiation spectra in eV units, and in XComp5r it is given in units of keV. This can be achieved by replacing all non-spectrum entries with the "#" symbol. Additionally, the last entry of the file must be a negative value, such as "-1," indicating the end of the spectra file for the reading code. Moreover, it is important to note that MC-GPU does not support energies below 12.5 keV or above 150 keV, so all corresponding energies outside this range need to be removed.

Thereby, for each irradiation event, the X-ray spectrum must be generated manually using the XComp5r software and then adapted for compatibility with MC-GPU. This process can be time-consuming, particularly for procedures involving a large number of events to simulate. Additionally, as explained in section 4.3, automating the X-ray spectrum generation is a fundamental requirement for an SDC (Skin Dose Calculation) tool. One potential solution would be to modify XComp5r as an extension of MC-GPU to enable automatic X-ray spectrum generation. However, since XComp5r is a closed-source code and cannot be modified, we sought an alternative approach.

To overcome this limitation, we developed a script based on the Spektr 3.0 software. Spektr 3.0 is a computational toolkit that calculates X-ray spectra using the Tungsten Anode Spectral Model using Interpolating Cubic Splines (TASMICS) algorithm. The toolkit includes a function library in Matlab (The MathWorks, Natick, MA) and an improved user interface (UI) (refer to Figure 17). Figure 17 demonstrates that Spektr 3.0 incorporates all the necessary parameters required by MC-GPU for X-ray spectrum generation. The X-ray spectrum generated by Spektr 3.0 can be exported to a .txt file, which can then be saved with the .spc extension required by MC-GPU.



Tube Select: Tube_01

kVp: 120

kV Ripple (%): 0

mm Cu: 1

mm Al: 2.5

Spectrum Model: ● TASMICS ○ TASMIP

Inherent Filtration: 3.6 / 0.005 mm of Al / W

Exposure: 1.568 mR/mAs at 100 cm from the source

Air Kerma: 1.3694E-02 mGy/mAs at 100 cm from the source

Fluence/Exposure: 272635 x-rays / mR / mm^2

Fluence/Air Kerma: 31216710 x-rays / mGy / mm^2

HVL: HVL1 Al 11.5967 mm

Mean Energy: 76.4581 keV

Plot Color: k

Hold On: ☑

Thickness: 0 mm

Equiv. mmAl: 0 mm

Thickness: 0 mm

Figure 17: Spektr 3.0 user interface.
`https://aapm.onlinelibrary.wiley.com/doi/10.1118/1.4955438`

The Spektr 3.0 has been modified and integrated in PyMCGPU-IR as a sub-routine. This sub-routine reads the needed information from the RDSR to automatically generate the X-ray spectrum file needed by MC-GPU using the Matlab functions library used by Spektr 3.0.

The original program Spektr 3.0 is validated against measurements and other existing models. In order to check if the modifications carried out to Spektr 3.0 does not affect the correct generation of the X-ray spectrum, a set of comparisons have been done with XComp5r.

Different set ups have been used to made the comparison by choosing values from 65 kV to 120 kV for the tube voltage, 0 to 4 mm for the Aluminium filter and 0 to 3 mm for the Copper, Beryllium and Tungsten filters. In this way, the most common X-ray sources in IR are covered. The results showed that the X-ray spectra generated by the sub-routine based on Spektr 3.0 and XComp5r are the same with only slight differences on the X-ray peaks. We have checked that these slight differences only introduce up to 1% differences in the dose calculation.

4.5 Table and mattress attenuation

In a typical IR procedure, the patient is positioned on a table or a mattress placed on top of the table. For simplicity, we will refer to both scenarios as "table." If the X-ray tube is positioned below the horizontal plane defined by the table, the X-ray photons emitted from the source will pass through the table before reaching the patient. This results in attenuation of the incident photons' energy. In addition, the skin dose is also affected by the forward scatter from the table [66].

To calculate the patient dose, it is essential to both consider the attenuation caused by the table and the forward scatter from the table. From now on, for the sake of simplicity, this two effects will be referred as table attenuation. Ideally, if we had precise information about the material composition and thickness of the table, we could directly simulate it as part of the voxelized geometry. However, manufacturers do not provide detailed specifications regarding the exact composition of the table.

The methodology employed in this study to account for table attenuation varies depending on the experiment but follows a consistent approach. It involves two steps. First, the table attenuation is determined through dose measurements. Second, the table is simulated as a layer of Aluminium with a thickness that reproduces the experimentally observed table attenuation. Different approaches have been used to measure table attenuation in various experiments, and these are described in the respective sections of Part IV.

4.6 Normalization factor

MC-GPU provides doses in the form of deposited energy per simulated history (eV/(ghistory)). To compare these simulated doses with absorbed doses, it is necessary to convert the deposited energy into absorbed dose using absolute values (Gy). This conversion is achieved by multiplying the deposited energy obtained from MC-GPU by a normalization factor, denoted as f. The normalization factor, f, is determined by the ratio between the experimental air kerma at a specific point and the deposited energy calculated by MC-GPU at that same point. The reference point is used for this purpose. Hence, the normalization factor, f, can be expressed as the ratio between the experimental air kerma ($K_{air,exp}$) and the simulated air kerma ($K_{air,sim}$) at the reference point:

$$f = \frac{K_{air,exp}}{K_{air,sim}} \tag{4.1}$$

The experimental air kerma ($K_{air,exp}$) value is derived from the kerma area product (KAP) obtained at the reference point. The KAP value is typically included in the radiation dose structured report (RDSR) and can be measured using a KAP-meter positioned on the gantry housing at a fixed distance from the focal spot. The reference point is a virtual position defined at a specified distance from the isocenter of the C-arm system. The field size at the reference point can be determined using the inverse square law, considering the source-to-detector distance and the collimated area at the detector, both of which are provided in the RDSR. To obtain the simulated air kerma ($K_{air,sim}$), a simulated box of air with dimensions 1 × 0.5 × 1 cm^3 is used.

4.7 From DICOM-RDSR to PyMCGPU-IR input files

As explained in section 4.3.3.2, PyMCGPU-IR uses the RDSR to obtain the necessary information for conducting simulations. Specifically, a set of parameters is extracted from the RDSR to calculate the required parameters for generating both the input and trigger files used by MCGPU-IR to compute the dose. The fields retrieved from the RDSR are itemized below, along with their corresponding definitions.

- Collimated (C)

  Area of the irradiated region in the detector.

- Dose Area Product (DAP)

  The DAP is defined as the absorbed dose multiplied by the area irradiated and is typically given in units of Gym^2. It is the absorbed dose measured in air and is measured by an ionizing chamber placed beyond the X-ray collimator. It is also sometimes called air kerma area product (KAP).

- Reference point

  Reference point defined by the angiographic system. Typically situated 15 cm below the isocenter, see Figure 27.

- Reference point air kerma ($K_{a,r}$)

  Cumulative air kerma at the reference point in units of Gy.

- Distance Source to isocenter (DSI)

  Distance from the source to the isocenter in units of mm.

- Distance Source to Detector (DSD):

  Distance from the source to the detector in units of mm.

- KVP:

  Peak kilovoltage output of the X-ray source in units of kV.

- X-Ray Filter material:

  Filter material in the X-ray tube.

- X-Ray Filter thickness:

  Thickness of the filter material in the X-ray tube.

- Positioner Primary Angle (PPA):

  The Primary Axis of rotation is defined at the intersection of the Patient Plane and of the Sagittal Plane. The Primary Axis is the x-axis in the MC-GPU system of coordinates (see Figure 23). The PPA is the angle of rotation in the plane perpendicular to the primary axis as shown in Figure 19. A positive angle corresponds to a rotation at the Patient left hand side (LAO) and a negative angle corresponds to a rotation at the Patient right side (RAO).

- Positioner Secondary Angle (PSA):

  The Secondary Axis is perpendicular to the Primary Axis and the Sagittal Plane as shown in Figure 20. The Secondary Axis is the y-axis in the MC-GPU system of coordinates (Figure 23). A positive angle corresponds to a rotation to the cranial direction (CRA) and a negative angle corresponds to a rotation to the caudal direction (CAU).

- Table longitudinal position (TLP):

  The direction of the longitudinal movement is perpendicular to the Primary Axis (x-axis) of rotation of the positioner. A positive value of the TLP indicates a movement towards the $+90°$ position of the positioner, see Figure 21. Therefore, a positive value indicates a movement in the negative direction of the x-axis.

- Table lateral position ($TLatP$):

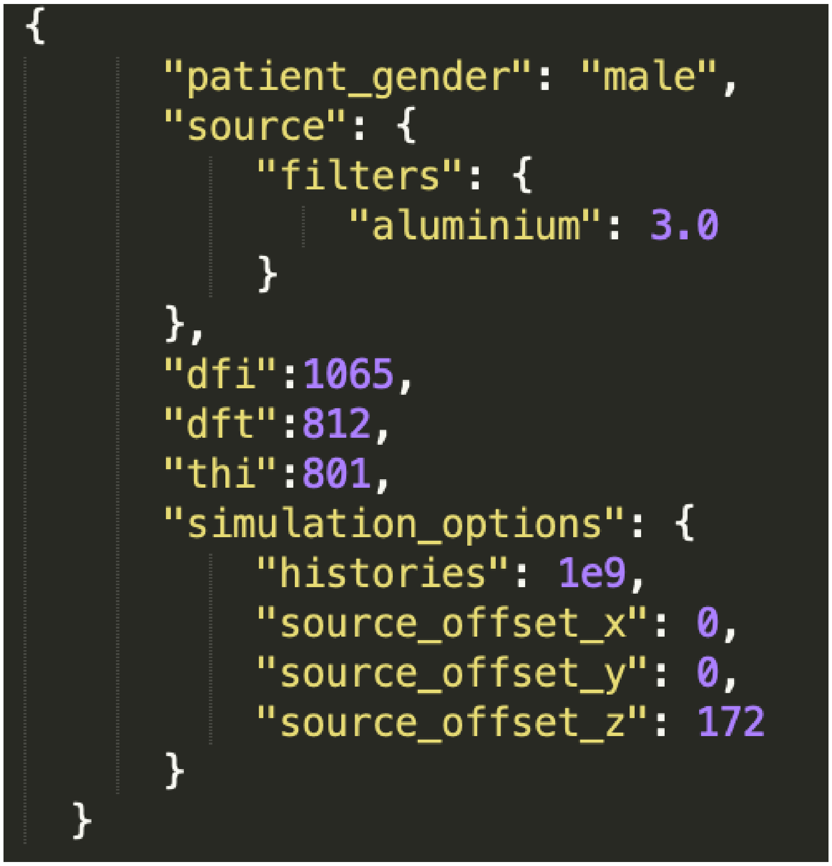
  The direction of the lateral movement is perpendicular to the Secondary Axis (y-axis) of rotation of the positioner. A positive value of $TLatP$ indicates a movement towards the $+90°$ position of the positioner, see Figure 22. Therefore, a positive value indicates a movement in the positive direction of the Z-axis.

- Table high position (THP):

  The THP position is the distance of the table respect to the isocenter. A positive value indicates that the table is moving below the isocenter towards the source.

At this point, we can address one of the significant challenges faced by SDC tools when it comes to automating dose calculations. The accurate calculation of dose and its location crucially depends on the relative position of the patient in relation to the radiation source, as this determines the specific region of the patient that is being irradiated. The RDSR provides us with only a few parameters, namely TLP, $TLatP$, and THP, that can be used to determine the patient's location. These parameters are provided in a specific coordinate system, with the default origin being the isocenter as defined above. However, the origin of coordinates can vary between procedures, and the new location may not be specified in the RDSR. In such cases, the SDC tools cannot accurately assign the patient dose.

Moreover, we have encountered cases where the THP parameter does not comply with the given definition, making it unclear from which point the value is given and how to interpret it. To address this issue, PyMCGPU-IR offers the ability to read an additional file alongside the input and trigger files, known as the complementary file, as mentioned in section 4.3.3.4. The complementary file is designed to complement the information obtained from the RDSR. The parameters included in the complementary file are displayed in the screenshot shown in Figure 18 and are detailed in Table 9.



```
{
    "patient_gender": "male",
    "source": {
        "filters": {
            "aluminium": 3.0
        }
    },
    "dfi":1065,
    "dft":812,
    "thi":801,
    "simulation_options": {
        "histories": 1e9,
        "source_offset_x": 0,
        "source_offset_y": 0,
        "source_offset_z": 172
    }
  }
```

Figure 18: Parameters of the complementary file

Table 9: Definition of the parameters of the complementary file.

Patient gender:

The gender of the patient is usually reported in the RDSR. If not, the advanced user can choose the gender of the patient by writing “male” or “female”. PyMCGPU-IR will choose the REX phantom for “male” and the Regina phantom for “female”.

Source:

Source information. Right now, only Aluminium inherent filtration can be defined (in mm).

Dfi:

Measured initial distance from floor to isocenter (in mm).

Dft:

Measured distance from floor to table (in mm).

Thi:

Table height displayed on the RDSR corresponding to the measured dft, namely, corresponding to the first event (in mm).

Simulations options:

Histories: number of photons to be simulated.

Source offset x: correcting value in the x axis (in mm).

Source offset y: correcting value in the y axis (in mm).

Source offset z: correcting value in the z axis (in mm).

The parameters dfi, dft, thi, and source offsets x, y, and z are used to accurately calculate the source position relative to the patient and take into account the various types of problems that can occur when extracting information from the RDSR. These parameters are specifically designed to address the challenges and discrepancies that may arise during the extraction process from the RDSR. Further details on these problems and their resolutions will be explained in section 4.7.7.

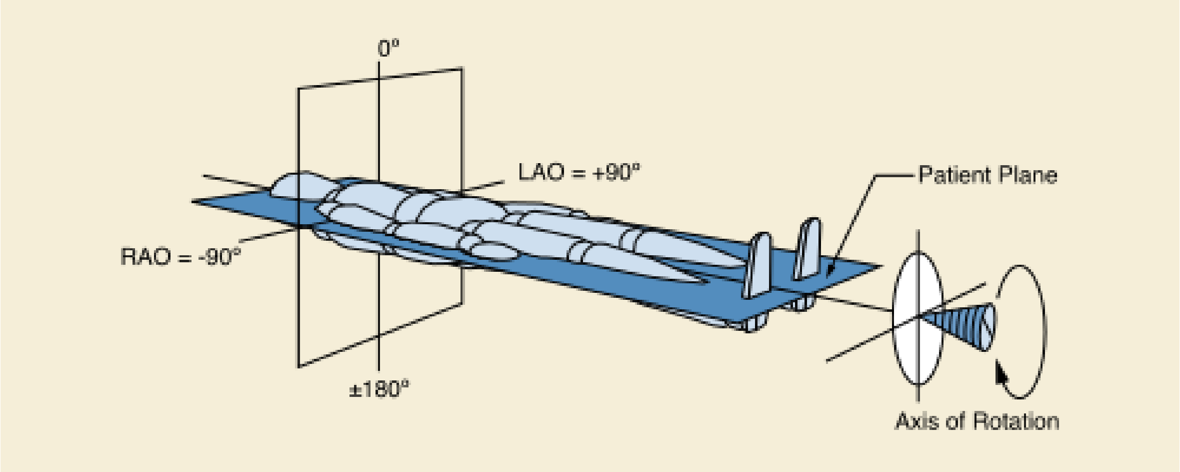


Figure 19: Position Primary Angle (PPA). Figure taken from [67]

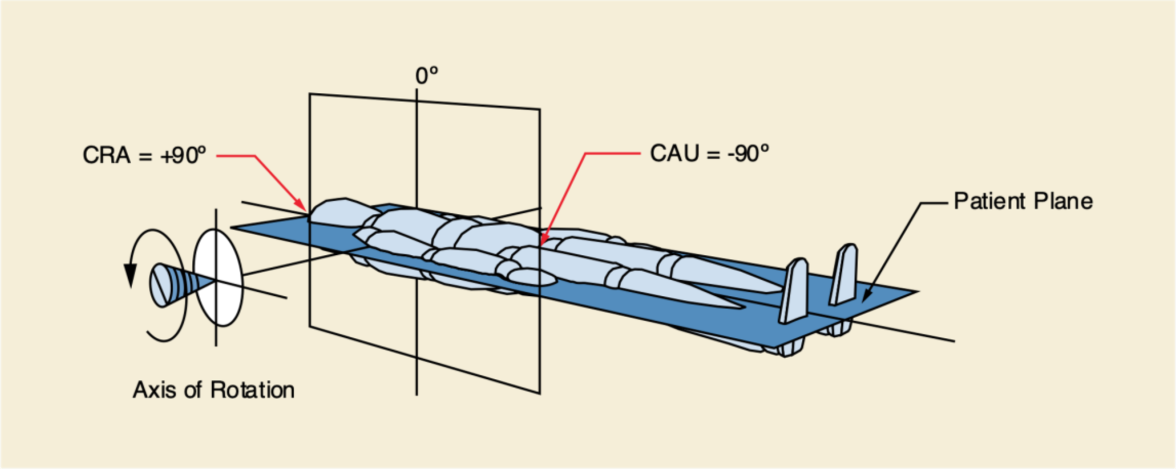


Figure 20: Position Secondary Angle (PSA). Figure taken from [67]

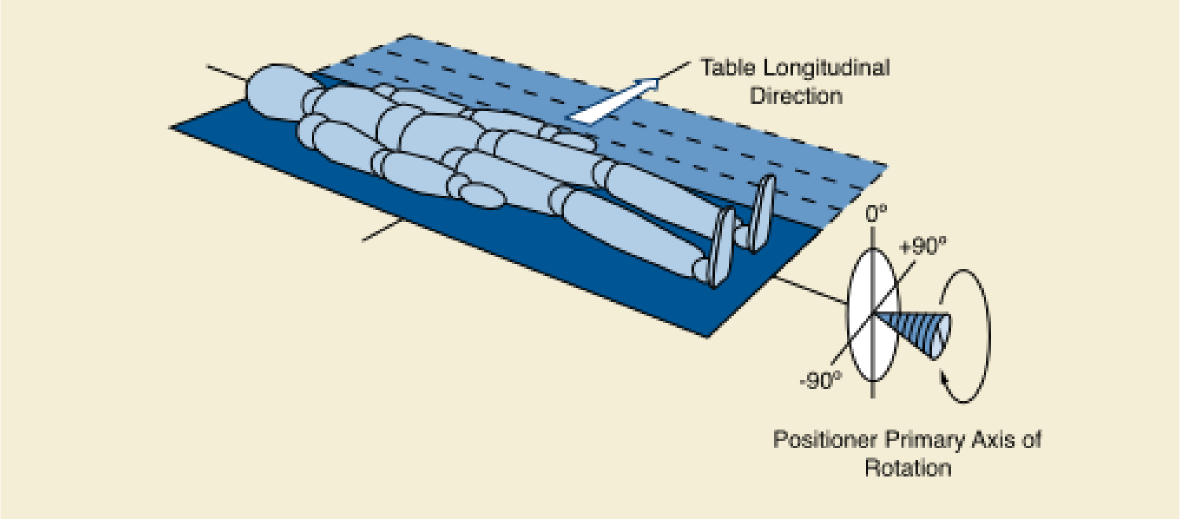


Figure 21: Table Longitudinal Position (TLP). Figure taken from [67]

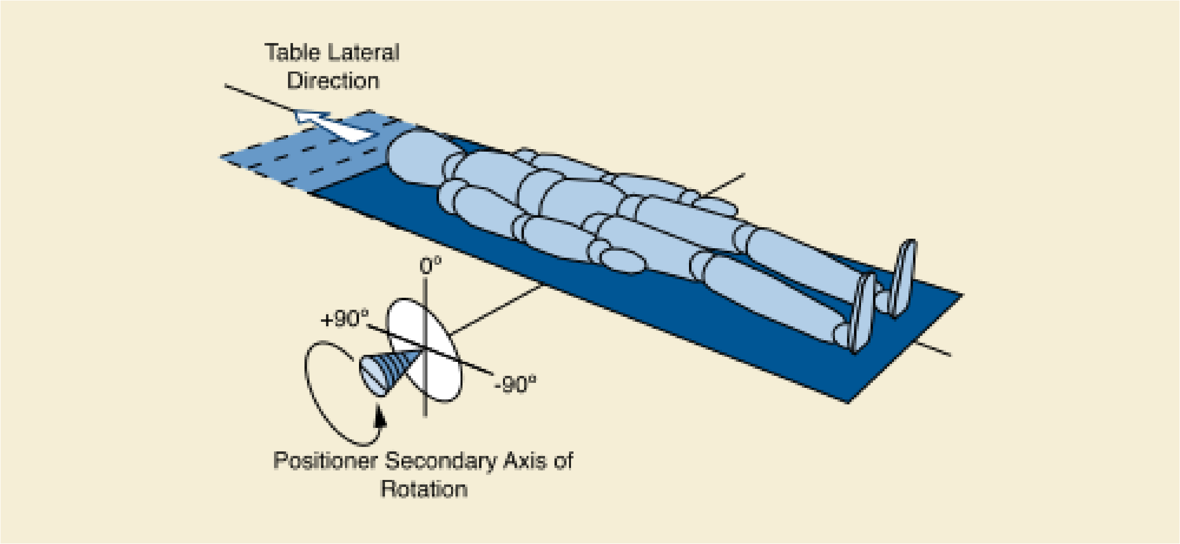


Figure 22: Table Lateral Position (TLatP). Figure taken from [67]

Figure 23 illustrates the system of coordinates used by MCGPU-IR. In this system, the origin of the coordinates is located at the lower corner of the patient's left foot. The x-direction is perpendicular to the sagittal plane of the patient, the y-direction is perpendicular to the patient plane, and the z-direction is perpendicular to both the x and y-directions. Consequently, when retrieving parameters related to the patient's location from the RDSR, it is necessary to transform them from the RDSR system of coordinates to the MCGPU-IR system of coordinates.

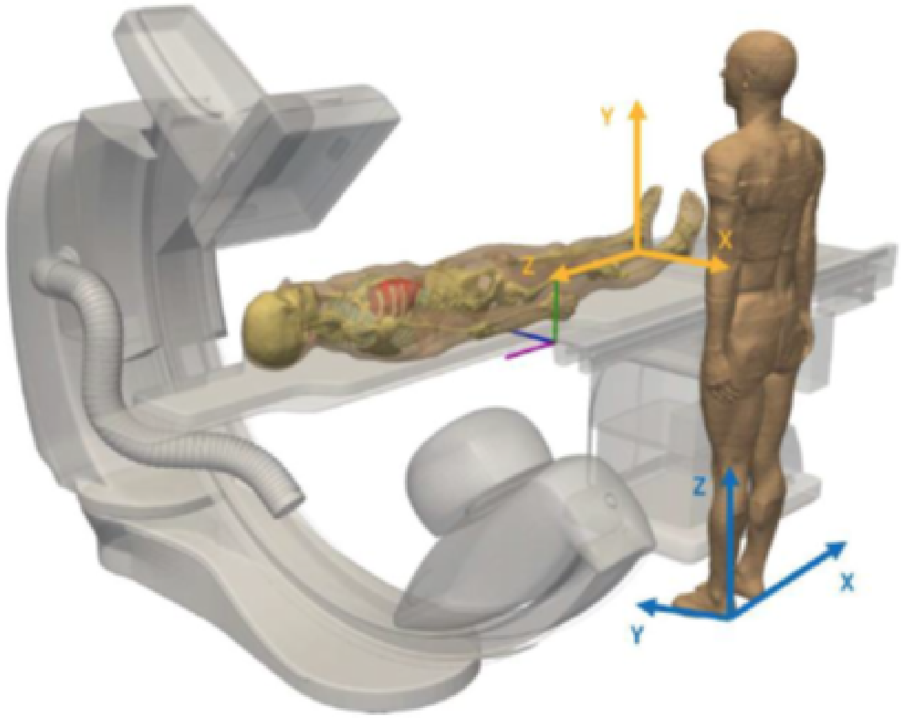


Figure 23: MCGPU-IR system of coordinates

As detailed in sections 4.2.1 and 4.3.1, all the parameters required by MCGPU-IR are organized into different sections within the input and trigger files. To calculate these parameters, PyMCGPU-IR uses the values retrieved from the RDSR, as presented in section4.7. In the following explanation, we will describe how PyMCGPU-IR calculates the parameters for each section of the input and trigger files based on the retrieved RDSR parameters.

4.7.1 Source section

Table 10: Parameters of the source section

X-ray energy spectrum file The X-ray energy spectrum file is generated using the sub-routine explained in section 4.4 using as input the fields KVP, X-Ray Filter Material and X-Ray Filter Thickness.
Source position Explained in 4.2.1.2.
Source direction cosines The values are always fixed to 0 1 0
Beam angular apertures The polar (ϕ) and azimuthal (ψ) angles are calculated as: • $\phi = 2\arctan\left(\frac{D_z}{2DSD}\right)$ • $\psi = 2\arctan\left(\frac{D_x}{2DSD}\right)$ where D_x and D_z are calculated as explained in Table 11.

4.7.2 Image detector section

Table 11: Parameters of image detector section

Output image fine name

Not used in PyMCGPU-IR

Number of pixels in the detector image

As the image is not used, the value is fixed.

Detector image size

D_x = SHUTTER.LEFT+SHUTTER.RIGHT and D_z = SHUTTER.BOTTOM+SHUTTER.TOP

or (in case that the shooter fields are missing in the DICOM)

$D_x = D_z = \sqrt{C}$

Source-to-detector distance

The DSD is already given by the RDSR.

4.7.3 CT scan trajectory section

Table 12: Parameters of the CT scan trajectory section

CT scan trajectory section

Not used in PyMCGPU-IR. The values are fixed, only one projection is done.

4.7.4 Dose deposited section

Table 13: Parameters of the dose deposited section

Tally material dose Values fixed to YES
Tally 3D voxel dose Values fixed to YES.
Output voxel dose file name Automatically chosen by PyMCGPU-IR.
Voxel Dose ROI X-index, Y-index and Z-index: automatically chosen by PyMCGPU-IR to cover all the voxelized phantom.

4.7.5 Voxelized geometry section

Table 14: Parameters of the volxelized geometry section

Voxel geometry file PyMCGPU-IR has the REX and REGINA phantoms to model a male and female patient, respectively. Depending on the gender introduced in the complementary file, PyMCGPU-IR will choose the REX or the REGINA phantom.

4.7.5.1 Material file list section

Table 15: Parameters of the material file list section

Material file list section Depending on the phantom chosen, in the voxelized geometry section, PyMCGPU-IR will automatically introduce all the materials of the corresponding phantom.

4.7.6 Trigger file

Table 16: Parameters of the trigger file

Source position
Same as in 4.2.1.2.
Source aperture angles
Same as in 4.2.1.2.
Source-to-rotation-axis
This is the distance source to isocenter (DSI), given by the RDSR.
Source rotations around x (lateral) and z (craneo-caudal) axis
This are the position secondary angle (PSA) and the position primary angle (PPA), given by the RDSR.
Conversion factor
This is the normalization factor explained in 4.6.
Energy spectrum file name
Same as in 4.2.1.2.

4.7.7 Calculation of the source position coordinate

The location of the source position with respect to the system of coordinates presented in Figure 23 is calculated using the parameters extracted from the RDSR together with the data in the complementary files following equations 4.2, 4.3 and 4.4.

$$x_{source} = \frac{x_{size}}{2} + TLP + source\_offset\_x \tag{4.2}$$

$$y_{source} = -(dft - (dfi - dsi) + (thp - thi) + source\_offset\_y) \tag{4.3}$$

$$z_{source} = \frac{z_{size}}{2} - TLatP + source\_offset\_z \tag{4.4}$$

where x_{size}, y_{size} and z_{size} are the dimensions of the voxelized phantoms in the x, y and z directions, respectively.

4.8 Uncertainty calculation

The MC-GPU program calculates two different types of uncertainties. Firstly, it computes the uncertainty associated with the averaged dose per voxel across multiple histories. Secondly, it determines the uncertainty related to the averaged dose per history among voxels that belong to a specific material or tag. To facilitate understanding, we introduce the following nomenclature:

Table 17: Different parameters definition

Parameter	Definition
$e_{i,j}$	Deposited energy in eV for history i and voxel j.
$q_{i,j}$	Dose in eV/g for history i and voxel j.
m_j	Mass of voxel j.
Q_j	Averaged dose among histories for voxel j.
u_j	Associated uncertainty of Q_j.
N	Number of histories.
Q_{mat}	Averaged dose in a given material or tag.
u_{mat}	Associated uncertainty of Q_{mat}.
K	Number of voxels corresponding to material or tag.

The MC-GPU program computes the deposited energy ($e_{i,j}$) in each voxel for each history. Then, the averaged dose (Q_j) among histories in each voxel is calculated as:

$$Q_j = \frac{1}{N}\sum_{i=1}^{N}\frac{e_{i,j}}{m_j} = \frac{1}{N}\sum_{i=1}^{N} q_{i,j} \tag{4.5}$$

where we have used equation 4.6

$$q_{i,j} = \frac{e_{i,j}}{m_j} \tag{4.6}$$

The uncertainty u_j associated to equation 4.5 is calculated as

$$u_j = \frac{1}{\sqrt{N}} \sqrt{\frac{1}{N m_j^2} \sum_{i=1}^{N} e_{i,j}^2 - \frac{1}{N^2 m_j^2} \left(\sum_{i=1}^{N} e_{i,j} \right)^2} \tag{4.7}$$

Using equation 4.6 we rewrite equation 4.7 as

$$u_j = \frac{1}{\sqrt{N}} \sqrt{\frac{1}{N} \sum_{i=1}^{N} q_{i,j}^2 - \frac{1}{N^2} \left(\sum_{i=1}^{N} q_{i,j} \right)^2} = \frac{1}{\sqrt{N}} \sigma_j \tag{4.8}$$

where σ_j is the standard deviation of the doses per history in voxel j. The averaged dose for a given material or tag (Q_{mat}) is calculated as:

$$Q_{mat} = \frac{1}{K} \sum_{j=1}^{K} Q_j \tag{4.9}$$

where K is the number of voxels belonging to a given material or tag. The uncertainty u_{mat} associated to equation 4.9 is given by the propagation of the uncertainties on the mean. Therefore, the uncertainty u_{mat} is given by

$$U_{mat} = \frac{1}{K} \sum_{j=1}^{K} u_j^2 \tag{4.10}$$

Using equation 4.8, we can rewritte equation 4.10 as

$$u_{mat} = \frac{1}{K} \sqrt{\sum_{j=1}^{K} \frac{1}{\sqrt{N}} \left[\frac{1}{N} \sum_{i=1}^{N} q_{i,j}^2 - \frac{1}{N^2} \left(\sum_{i=1}^{N} q_{i,j} \right)^2 \right]} = \tag{4.11}$$

which can be rewritten as

$$u_{mat} = \frac{1}{K\sqrt{N}} \sqrt{\frac{1}{N} \sum_{j=1}^{K} \sum_{i=1}^{N} q_{i,j}^2 - \frac{1}{N} \sum_{j=1}^{K} \left(\sum_{i=1}^{N} q_{i,j}^2 \right)^2} \tag{4.12}$$

The uncertainties mentioned are directly computed by MC-GPU and represent the statistical uncertainty associated with the simulation, expressed in eV/g. However, when converting the results to Gy using the normalization factor (discussed in section 4.6), the uncertainty of the KAP measurement needs to be taken into account. In this work, the uncertainty associated with the KAP measurements is considered to be 10% for a coverage factor $k = 2$.

Chapter 5

Dose measurements

5.1 Dose measurements for the MC-GPU validation

5.1.1 Thermoluminescence dosimetry system

Measurements were performed at the personal dosimetry service of UPC (LTLD). The detectors were thermoluminescent dosimeters (TLDs) LiF: Mg,Cu,P crystals manufactured by Conqueror Electronics Technology Co. Ltd. (Beijing, China) under the trade name TLD-2000C. They are 4.5- mm-diameter circular pellets, 0.8 mm thick, with a mass density of 2.65 g/cm^3. The annealing process previous to the detectors exposure to radiation was performed in a PTW TLDO oven, and consisted of 240 ºC for 10 min followed by a fast quench on an Aluminium block. The readout after the exposure was done on a semiautomatic Harshaw 5500 reader (Thermo-Electron) in a two-step heating cycle: a preheat process at a temperature of 160 ºC for 10s, followed by a reading phase of 26 s up to 250 ºC with a linear heating rate of 4 ºC/s. The readout was carried out within 24h after irradiation. Instead of using batch calibration, detectors were individually calibrated. To determine the response of a single detector with respect to the whole set, individual calibration factors were determined by irradiating all detectors using a ^{137}Cs source, at a value of air kerma of 3 mGy. The use of an individual correction factor is a standard procedure to reduce uncertainty in the measurements [68]. Several background dosimeters were used for each measurement and the background signal was subtracted from the measured reading. In order to provide the output in terms of absorbed dose (Gy), calibration factors were calculated in order to convert reader arbitrary units to air kerma (Gy) in the material in which the detectors are embedded. It is worth mentioning that in the energy range of interest differences between air kerma and absorbed dose in air are negligible. These calibration factors were determined under reference conditions at the UPC facility.

5.1.2 Energy calibration for the TLDs used in the hospital

To obtain appropriate calibration factors for measuring air kerma with TLDs in the hospital setting, the energy calibration of the dosimetric system was conducted at the UPC calibration laboratory (LCD). A group of 10 dosimeters was irradiated at the laboratory, which holds accreditation from the Spanish Accreditation Body (ENAC) and traceability to the PTB (German National Metrology Institute) for IEC61267 qualities [22]. The dosimeters were exposed to two IEC61267 qualities, namely RQR5 and RQR6, which were chosen due to their similar X-ray characteristics to the beams encountered in the hospital [69]. Table 18 presents the key features of these beams.

Table 18: Features of the beam qualities used for irradiation under controlled conditions at the UPC laboratory

Beam quality	Anode angle	Voltage (kV)	Inherent filtration	Added filtration	1 st HVL
RQR5	18º	70	7 mm Be	2.5 mm Al	3.07 mm Al
RQR6	18º	80	7 mm Be	3.0 mm Al	3.45 mm Al

5.1.3 Uncertainty of TLD measurements

The uncertainty associated with TLD measurements used in UPC calibration laboratory (7.2.1.1) corresponds to the combination of the standard deviation of the different dosimeter readings and the uncertainty associated with the experimental set-up, and is of the order of 5%, with a coverage factor, $k = 2$.

When the TLD readings are corrected by the energy calibration factor described in 5.1.2, such as in the measurements in a hospital setup 7.2.3.1, an additional uncertainty component associated to the difference between the calibration beams and the hospital beams, as well as the reference air kerma uncertainty, must be taken into account. Thus, in this case, the uncertainty associated to the measurements is higher than the TLD uncertainty of the measurements for the LCD setup. An uncertainty of 8% ($k = 2$) is assigned to the TLD measurements at the Hospital Vall d'Hebron study.

5.2 Dose measurements for the validation and implementation of PyMCGPU-IR

The experimental dose assessment involved the use of TLDs obtained from the SCK CEN dosimetry service [70]. These TLDs were cylindrical pellets with a thickness of 0.9 mm and a diameter of 4.5 mm, composed of LiF:Mg,Cu,P material (MCP-N) from the Institute of Nuclear Physics, Poland. Before irradiation, the TLDs underwent an annealing process at 240 ºC for 10 minutes, followed by rapid cooling in a freezer at 10 ºC. Prior to read-out, the dosimeters were heated at 120 ºC for 30 minutes. The read-out of the TLDs was performed using a Harshaw 5500 system, employing a constant heating rate of 10 ºC/s from room temperature up to 240 ºC. The readings were adjusted for detector sensitivity, which was determined individually using a ^{137}Cs source. Calibration of the TLDs took place at a primary standard laboratory, employing a heavily filtered beam that represented interventional radiology conditions (80 kVp, 0.9 mm Cu added filtration, and 8.63 mm Al HVL), as described in [71].

The uncertainty associated with the TLD measurements was provided by the SCK·CEN service and is estimated to be 20% for a coverage factor $k = 2$. This uncertainty accounts for various factors, including the characteristics of the TLD (such as energy and angle dependence), their calibration, and the reading system as described in [72] for exposure conditions encountered in IR.

Furthermore, Gafchromic films (25.4 × 30.48 cm^2, XR-RV3, Ashland, USA) were used in certain procedures to measure the skin dose distribution.

Chapter 6

Computational resources

The simulations presented in this study were performed on the UPC Argos Cluster. PENELOPE/penEasy was executed on any of the 15 available nodes and used two Intel Xeon E5520 Processors, 8x2GB Dual Rank UDIMMs at 1066 MHz, and a 160 GB SATA 7200 3.5" hard drive. MC-GPU and PyMCGPU-IR, on the other hand, made use of GPUs and ran on a single node with compatible NVIDIA graphics cards. Specifically, the setup included two NVIDIA GeForce 1080Ti GPUs with 11GB VRAM each, an Intel Xeon E5-2670 v3 CPU, 4 x 16GB RDIMM at 2133MHz. The CUDA version used was 8.0, and the MPI version employed was 1.10.7. It should be noted that MC-GPU and PyMCGPU-IR can be executed on other machines compatible with MC-GPU, such as laptops and cloud-based systems. The PENELOPE/penEasy code was compiled using the gfortran 4.8.5 compiler. PyMCGPU-IR can be distributed as a binary application. To use it, one needs to upload a copy of MCGPU-IR (and extended version of MC-GPU that will be presented on section 4.3.1) and ensure that the requirements for MCGPU-IR are met. If using the source code, the main requirement is to have Python 3.6 or a newer version installed. The complete list of requirements and instructions for using PyMCGPU-IR from the source code can be found in its user manual [73].

Part IV

Results

Chapter 7

MC-GPU validation

The validation of MC-GPU was conducted in two steps. Firstly, MC-GPU was compared with a well-validated multipurpose Monte Carlo (MC) simulation code, namely, PENELOPE/penEasy. Secondly, experimental measurements were performed in a calibration laboratory and under realistic conditions in an operating room, and these measurements were then compared with MC-GPU simulations.

7.1 Comparison against PENELOPE/penEasy

The code package used for comparison with MC-GPU was PENELOPE (version 2014)/penEasy (version 2015). PENELOPE is widely recognized as a reference code for Monte Carlo simulation of electron, positron, and photon transport. penEasy [74] is a specific main program designed for PENELOPE. It provides the user with a set of source models and tallies that can be invoked from a structured code.

7.1.1 Simulated setup for the comparison with PENELOPE/penEasy

The Duke phantom was used for comparison purposes (see subsection 4.1.4). The doses were calculated at two levels: at the whole organ or tissue level (comprised of thousands of voxels) and at the individual voxel level, and then compared. In both programs, the dose in each material is calculated as the average of the dose across all voxels belonging to the same material. Additionally, both programs provide the maximum dose value and its corresponding location.

Due to the specific energy range and computational speed considerations, MC-GPU does not simulate the transport of charged particles. To facilitate comparison with PENELOPE, the energy absorption parameters of electrons and positrons in PENELOPE/penEasy were set to very high energies (e.g., 1E9

MeV). By doing so, once the secondary particles are generated, they are locally absorbed in the material and no longer tracked. This approximation is valid for the energy range relevant to interventional radiology [75, 76, 77]. The requested outputs for the PENELOPE/penEasy code included the dose in each voxel and the energy deposited in each labeled material. Doses were provided in eV/(g·history) (eV/g per simulated history), while energy deposition units were in eV/history. The maximum dose value and position were easily obtained from the dose distribution. Furthermore, the dose in each tissue could be directly calculated by dividing the deposited energy by the corresponding tissue mass.

The comparison employed a typical diagnostic X-ray spectrum defined in the International Standard IEC 61267 [22]. This spectrum represents one of the reference radiation qualities in the field of X-ray diagnostics for radiation emitted by X-ray sources. The calculations were performed for RQR6 quality [22], generated by a tube voltage of 80 kV (see details in Table 18). The X-ray energy distribution for the simulations was generated using XCOMP5R (see section 4.4). The X-ray source was defined to irradiate the chest of the phantom completely. A postero-anterior (PA) projection was simulated, ensuring that the radiation strikes the back of the phantom. The Duke phantom was positioned at a distance of 30.5 cm from the beam source. To fully irradiate the chest of the Duke phantom, a pyramidal field with dimensions of 61 x 61 cm^2 was defined at a distance of 30.5 cm from the source. The 61 cm width corresponds to the width of the Duke phantom. This beam size was deliberately chosen to encompass all the different organs within the chest of the Duke phantom, thus it is significantly larger compared to the beam sizes typically employed in interventional radiology, which aim to irradiate specific organs.

7.1.2 Results of the comparison with PENELOPE/penEasy

Figure 24 and Table 19 present the dose distribution for chest irradiation of the anthropomorphic phantom and the average dose values for each tissue, respectively [78]. The codes demonstrated excellent agreement, with differences below 1%. These small differences can be attributed to the distinct algorithms used by the codes to calculate photon cross sections. PENELOPE v2014 employs tabulated photon cross sections on a denser energy grid, allowing for a more detailed representation of the cross section near photoelectric absorption edges, and interpolates them on a linear log-log scale. In contrast, MC-GPU uses tabulated values on a linear grid.

The execution time for the PENELOPE/penEasy code was approximately 9.3 hours to complete the simulation and achieve standard errors in the materials of about 0.1% ($k = 2$). On the other hand, MC-GPU required only 13 seconds for execution, reaching standard errors of a similar order while being approximately 2500 times faster than PENELOPE/penEasy. Furthermore, both programs identified the maximum dose voxel at the same position, namely, in the air at the patient's entrance. High doses were also observed in the bone and on the skin on the most exposed side (the patient's back). The difference in the reported Peak Skin Dose (PSD) between the two codes was merely 0.6%, with an associated statistical uncertainty ($k = 2$) of 1% for MC-GPU and 2% for PENELOPE/penEasy.

Table 19: Type of tissue, number of voxels, computed dose from PENELOPE/penEasy and MC-GPU, uncertainty of the simulation (U) with ($k = 2$), ratio between the two codes for the anthropomorphic phantom simulation and uncertainty of the ratio (U_{ratio}) [78]

Material/Tissue	#voxels	PENELOPE dose eV/(g·history)	U (%)	MC-GPU dose eV/(g·history)	U (%)	$\frac{\text{MC-GPU}}{\text{PENELOPE}}$	U_{ratio} (%)
Air	2273413	1.38E-01	0.16	1.38E-01	0.16	1.00	0.23
Muscle	278365	2.27E-01	0.01	2.27E-01	0.01	1.00	0.02
Soft tissue	34535	1.37E-01	0.05	1.38E-01	0.04	1.01	0.07
Bone	31577	6.30E-01	0.02	6.32E-01	0.02	1.00	0.02
Cartilage	11840	6.34E-02	0.12	6.39E-02	0.11	1.01	0.16
Adipose	104890	1.66E-01	0.02	1.66E-01	0.02	1.00	0.03
Blood	8945	1.07E-01	0.10	1.07E-01	0.10	1.00	0.15
Skin	39449	2.68E-01	0.03	2.67E-01	0.03	1.00	0.04
Lung	30794	2.93E-01	0.04	2.95E-01	0.04	1.01	0.06

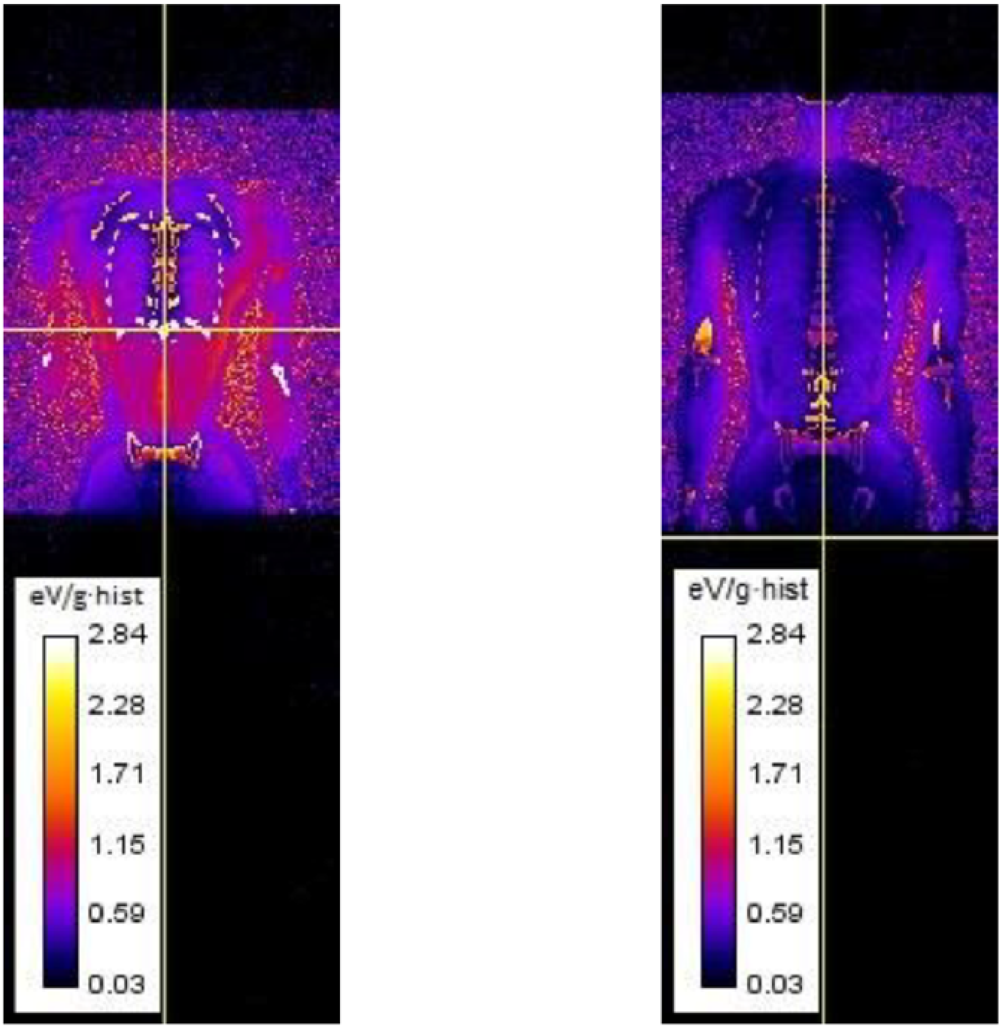
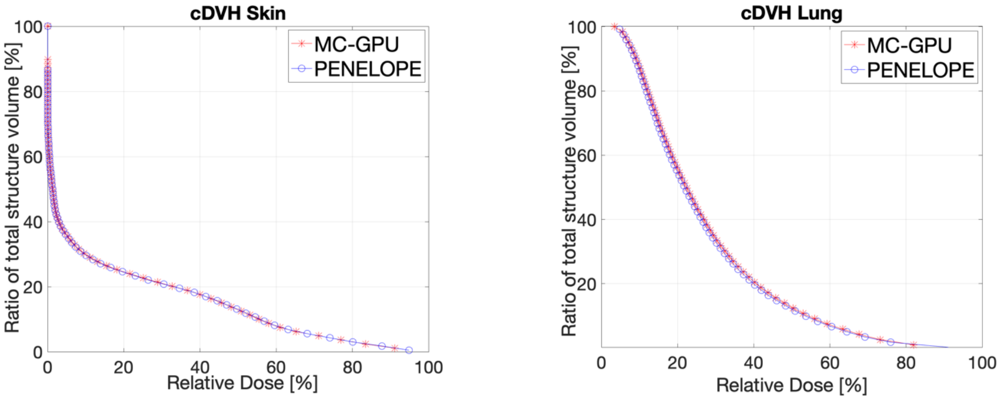


Figure 24: Voxel dose distribution for MC-GPU for the postero-anterior irradiation of the Duke phantom. (a) back of the Duke and (b) front of the Duke

To assess the consistency of the dose distribution across all voxels between the two codes, dose-volume histograms (DVH) were generated for each tissue. DVHs provide a graphical representation of the relationship between radiation dose and tissue volume, summarizing 3D dose distributions in a 2D format (Figure 25). In cases where a material is distributed throughout the human body, such as the skin, the voxel dose varies across a wide range due to variations in exposure (e.g., back skin directly exposed to the radiation beam versus front skin). This variability makes it challenging to discern differences between the two code outputs. However, this is not the case for tissues associated with specific organs, such as the lungs. Figure 25 illustrates the cumulative DVHs for the skin and lungs, demonstrating excellent agreement between MC-GPU and PENELOPE.



(a) skin

(b) lung

Figure 25: Cumulative Dose-Volume Histograms (cDVH) for skin (left) and lung (right)

7.2 Comparison against measurements

Various experimental measurements were conducted at two different locations: the UPC Calibration Facility and an operating room at the Hospital Universitari Vall d'Hebron (HUVH) hospital. These measurements involved the use of Thermoluminescent Detectors (TLDs) from the LTLD.

Multiple setups were tested, using slab phantoms with surface dimensions of 30x30 cm^2 and constructed from various tissue-equivalent materials. Some of these plates included small cavities where the detectors could be placed. The materials used for the plates included plastic water (PW®, ρ=1.03 g/cm^3), lung tissue-equivalent material (ρ=0.199 g/cm^3), and bone tissue-equivalent material (ρ=1.93 g/cm^3) from Computerized Imaging Reference Systems (CIRS). The lung and bone equivalent materials closely matched the linear attenuation coefficients of their respective reference tissues [79] within a range of ± 2.5%. The materials were simulated based on the atomic composition specified by the manufacturer. At each measurement position, nine TL detectors were exposed to radiation.

7.2.1 UPC Calibration Laboratory measurements

7.2.1.1 Experimental setup

The initial validation process took place in a controlled environment at the UPC Calibration Laboratory. The irradiations were performed using sandwich-like phantoms constructed from the above-mentioned tissue-equivalent materials. TLD detectors were positioned at various depths (as specified in Table 20), relative to the point of entry for photons into the phantom. The irradiations were conducted using RQR5 and RQR6 qualities, as indicated in Table 18. For all cases, the delivered air kerma was set at 5 mGy at a distance of 2 m from the focal spot. The radiation field at the irradiation plane (defined as the plane where the photons enter the material) had a diameter of 32 cm.

Table 20: Phantom thickness and dosimeter position inside the phantom for measurements under controlled conditions at the UPC laboratory.

Phantom	Total thickness (cm)	Depth (cm)
PW®	1.5	0.5
Lung	4	2
Bone	2	1

Additionally, irradiations were conducted in free air conditions. For these irradiations, the TLD detectors were enclosed within small opaque polyethylene plastic bags, which had a density thickness of 11 mg/cm^2. Figure 26 provides a schematic representation of the irradiation configurations used at the UPC Calibration Laboratory.

Figure 26: Irradiation configuration at the UPC laboratory

Three voxelized geometries were implemented in MC-GPU to replicate the PW®, lung, and bone phantoms. Each voxel had a size of 3.33 x 3.33 cm^2 in the plane perpendicular to the beam and a thickness of 0.1 cm in the direction of the beam. The experimental setup included nine detectors arranged over an area of 3.33 x 3.33 cm^2. Therefore, the TLDs were modeled as a single voxel with a thickness of 0.1 cm, representing air. To simulate the irradiation of the TLDs in free air, without the presence of attenuation or scattering materials, the same geometries were used, but the phantoms were treated as being entirely composed of air. The calculated dose in the corresponding single voxel was compared to the average of the nine dose measurements.

7.2.2 Results of the UPC calibration laboratory measurements

Table 21 presents the results of the measurements and simulations, given as the ratio of the detector readings within different materials (L_{mat}) to the reading in air (L_{air}), for both RQR5 and RQR6 qualities. These results correspond to the experiments conducted at the UPC Calibration Laboratory [78]. The simulation time (t_{sim}) and the ratio between the experimental and simulation results are also provided, along with the expanded uncertainty of the ratio for a coverage factor ($k = 2$). The uncertainty associated with TLD measurements encompasses the standard deviation of the dosimeter readings and the uncertainty related to the experimental setup, in the order of 5%. The simulations were performed using MC-GPU, generating 5E10 histories, with an execution time of approximately 1.5 minutes and standard deviation ranging from 2% to 4% (2 sd). As the data are normalized to air kerma, only the statistical uncertainty is considered in the calculation for this verification. The expanded uncertainty of the ratio (U_{ratio}) is determined by combining the experimental and Monte Carlo uncertainties (for $k = 2$).

Table 21: Experimental and simulated ratios $\mathrm{L}_{mat}/\mathrm{L}_{air}$ for PW®, Lung, Bone equivalent phantoms and RQR5 and RQR6 qualities. Time of simulation t_{simul} (s) and ratio between the experimental and simulated output with associated expanded uncertainty

Phantom	Radiation Quality	TLD	Simulations(5E10 histories)			
		$\mathrm{L}_{mat}/\mathrm{L}_{air}$	$\mathrm{L}_{mat}/\mathrm{L}_{air}$	T_{simul}(s)	Sim/TLD	U_{ratio}
PW	RQR5	0.953	0.975	70	1.02	5 %
Lung	RQR5	1.087	1.083	46	1.00	6 %
Bone	RQR5	0.279	0.270	90	0.97	7 %
PW	RQR6	1.070	1.024	71	0.96	6 %
Lung	RQR6	1.122	1.058	47	0.94	6 %
Bone	RQR6	0.367	0.366	97	1.00	7 %

Column 7 of Table 21 displays the ratios between the simulated and experimental values. Generally, the differences are below 5%, except for the lung with RQR6, where a difference of 6% was observed. Additional simulations were conducted with 2E11 histories. Despite the longer execution time of up to 7 minutes and a standard deviation below 2%, there was no significant improvement compared to the simulations presented in Table 20. Therefore, the faster simulations were preferred due to computational time savings. Importantly, all the differences observed fall within the statistical expanded uncertainty of the ratio, U_{ratio}.

7.2.3 Hospital measurements

7.2.3.1 Experimental setup

The second part of the measurements took place at the Department of Hemodynamics in the HUVH hospital, using a real operating room equipped with a Philips Allura Clarity X-ray system. Figure 27 illustrates the configuration used for irradiation at the HUVH hospital. To simulate thorax irradiation during an interventional procedure, the phantom was composed of three different materials arranged in the following order from bottom to top: 2 cm of PW®, 2 cm of bone, 9 cm of lung, and 2 cm of PW®(Figure 28). This particular configuration is referred to as "phantom 1." The TLD detectors were positioned at depths of 1 cm, 3 cm, and 6 cm within the PW®, bone, and lung layers, respectively. Additionally, a similar irradiation was conducted using a phantom consisting of PW® layers only. In this case, the detectors were placed at depths of 1 cm, 7 cm, and 13 cm (28). This configuration is denoted as "phantom 2"

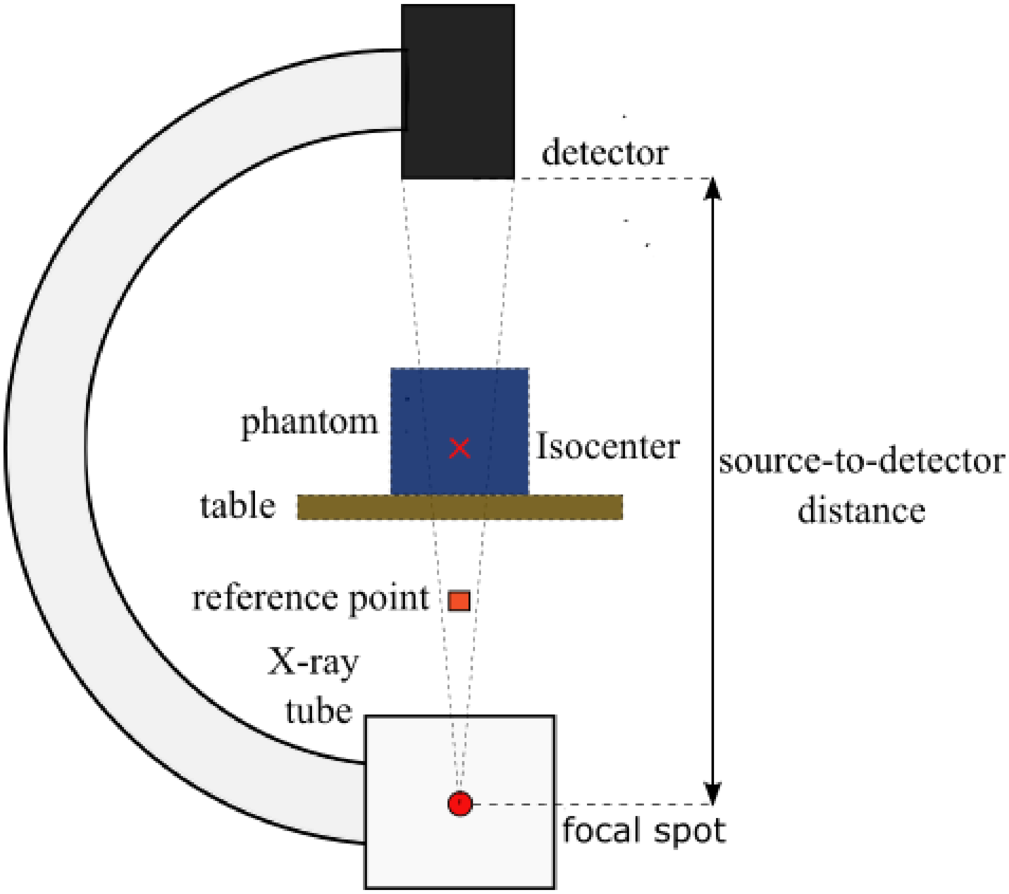
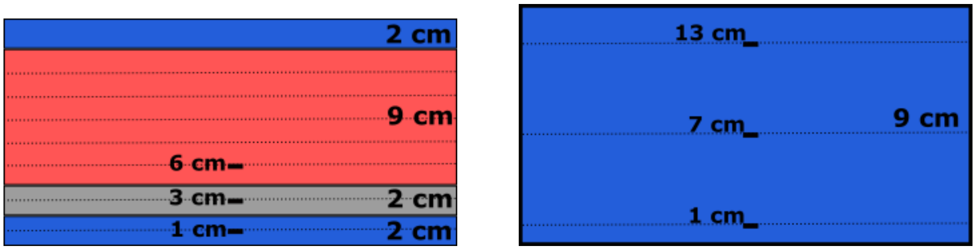


Figure 27: General scheme of the experimental set up for the measurements carried out in the operating room of the HUHV hospital.



(a) Phantom 1

(b) Phantom 2

Figure 28: (a) Phantom 1. Three different materials: PW® in blue, bone in gray and lung in red. The detectors are represented with black squares and are placed at 1, 3 and 6 cm depth in PW®, bone and lung plates, respectively. (b) Phantom 2. Made of PW® layers. The detectors are represented with black squares and are placed at 1, 7 and 13 cm depth.

Simulations were designed and conducted to replicate the two experimental configurations. Each set of nine TLDs at every depth level was modeled as a single voxel composed of air. Independent simulations were performed for each depth and material.

For the purpose of validation, it is essential to closely mimic the realistic scenario in the Monte Carlo simulation geometry. In clinical practice, the table pad on which the patient lies attenuates the beam intensity for under-table projections, such as the postero-anterior (PA) projection. To quantify this attenuation, two measurements were conducted in the clinical room using a semiconductor detector, the RaySafe X2. The detector was first placed on the table and then, under the same irradiation conditions, beneath the table. The ratio of the two measurements, expressed in terms of dose rate, provided the attenuation caused by the treatment couch. Monte Carlo simulations were performed to reproduce these two experiments, with and without the table, in order to calculate the equivalent thickness of Aluminuim (Al) required to produce the same attenuation. Based on the results, a 1.8-mm Al layer was incorporated into the voxel simulation geometry of the phantoms.

Irradiations were carried out for the PA projection under the conditions specified in Table 22. The radiation field size was 25 cm x 25 cm, and the distance between the source and the image detector was 100 cm.

Table 22: Features of the radiation beams used for irradiations with phantom 1 (Figure 28a) and phantom 2 (Figure 28b) under clinical conditions.

Phantom type	Anode angle	Voltage (kV)	Inherent filtration	Added filtration
Phantom 1	11º	62	2.5 mmAl	1 mm Al + 0.1 mm Cu
Phantom 2 (PW®)	11º	65	2.5 mmAl	1 mm Al + 0.1 mm Cu

7.2.3.2 Results of the hospital measurements

Table 23 presents a comparison between the measurements conducted at the hospital and the corresponding simulations performed on MC-GPU using 1E10 histories [78]. The TLD readings are adjusted with the energy calibration factor described in subsection 5.1.2. Consequently, the uncertainty in the measurements is higher (8%) compared to the uncertainty mentioned in section 7.2.1.1. In addition to the uncertainties considered in Table 21, factors such as the disparity between the calibration beams and the beams used in the hospital, as well as the uncertainty in the reference air kerma, must be taken into account.

The simulation data are presented in terms of absorbed dose in air, normalized by the factor N defined in equation 4.1. Therefore, the uncertainty associated with the simulated values incorporates both the statistical standard deviation and the uncertainty related to the Kerma-Area Product (KAP) recorded in the Radiation Dose Structured Report (RDSR). Column 8 displays the ratio between the simulated and measured doses, while the last column presents the expanded uncertainty associated with the ratio. The results indicate good agreement between the simulated and experimental doses, with differences below 15% and within the corresponding uncertainty range.

Table 23: Comparison of experimental and simulated absorbed dose in a clinical setup (expanded uncertainties are given for $k = 2$)

Phantom	Material	Experimetal		Simulated		t_{simul}	$\frac{\text{Simulated}}{\text{Experimental}}$	U_{ratio}
type	(depth)	D(mGy)	U(%)	D(mGy)	U(%)	(s)		(%)
1	PW®(1 cm)	2.61	8%	2.46	10%	37	0.95	13%
1	Bone (3 cm)	0.67	8%	0.70	11%	37	1.06	13%
1	Lung (6 cm)	0.21	8%	0.24	12%	37	1.12	14%
2	PW®(1 cm)	4.44	8%	3.99	10%	55	0.90	13%
2	PW®(7 cm)	1.30	8%	1.24	12%	55	0.95	14%
2	PW®(13 cm)	0.18	8%	0.19	16%	55	1.07	18%

Chapter 8

PyMCGPU-IR validation and implementation

In Section 4.3, it was mentioned that PyMCGPU-IR incorporates optimization functions and a clustering algorithm to enhance computational time performance and reduce the number of events to be simulated. Before implementing these features, various checks were conducted to ensure their effectiveness. The results of these checks are presented in Sections 8.1 and 8.2, for both the optimization functions and clustering algorithm, respectively.

The validation and implementation of PyMCGPU-IR were carried out through three progressive steps, each increasing in complexity. The first step involved using PyMCGPU-IR to replicate a set of measurements on PMMA phantoms. In the second step, PyMCGPU-IR was used to reproduce skin dose measurements for various beam angulations on an adult Rando Alderson anthropomorphic phantom. Lastly, in the third experiment, PyMCGPU-IR was employed to replicate organ dose measurements in three clinical procedures using the Rando Alderson phantom. The outcomes of these experiments are presented in Sections 8.3, 8.4, and 8.5, respectively.

8.1 Optimization functions

PyMCGPU-IR is an application written in Python for MCGPU-IR. In the past, when using MC-GPU, users had to manually enter two GPU performance-related parameters: the number of threads per block (also known as block size) and the number of histories per thread (known as thread size), as described in subsection 4.1.2. This process required users to have a basic understanding of CUDA programming in order to achieve optimal GPU performance. However, thanks to the newly developed functions, these parameter values are now automatically calculated based on the specific model of GPU in use, thereby eliminating the need for manual input.

Several variables play a role in determining the time performance, including the number of blocks per grid (grid size), the number of threads per block (block size), shared memory, and the number of registers (all of which are related to memory), as well as the number of histories per thread (thread size) (as discussed in subection 4.1.2). One commonly employed approach to determine suitable grid and block sizes is to aim for high occupancy. Occupancy is defined as the ratio of active warps per multiprocessor (Active Warps) to the maximum number of warps that can be active on the multiprocessor (Max Active Warps), as shown in Equation 8.1.

$$\text{Performance} = \frac{\text{Active Warps}}{\text{Max Active Warps}} \tag{8.1}$$

The number of active warps is influenced by the shared memory and the number of registers. The shared memory refers to the memory allocated for the variables declared within the code and has a fixed value of 47904 bytes in our case. On the other hand, the number of registers can be specified by the compiler, with higher values generally leading to better performance for individual GPU threads. However, increasing the number of registers also reduces the maximum block size and, consequently, the number of threads executed in parallel. For our specific code, we have determined that using 72 registers per thread yields the best results.

Once MC-GPU is compiled, both the shared memory and the number of registers are fixed. The remaining variables affecting time performance are the grid, block, and thread sizes. In MC-GPU, the block and thread sizes are manually entered by the user, with recommended values of 128 and 100, respectively. The grid size is heuristically computed by MC-GPU based on three GPU characteristics: the number of GPU cores, the core frequency, and the major CUDA compute capability.

The optimization functions automatically calculate the optimal grid, block, and thread sizes. These functions first determine the grid and block sizes that maximize occupancy, using functions provided by newer CUDA versions. An extract of the code (Figure 29) showcases the calculation of the optimal grid and block sizes. Once the total number of threads is obtained, the number of histories per thread is defined as the total number of histories divided by the total number of threads. This ensures that the CPU calls the GPU only once, minimizing time consumption.

Simulations were conducted for various setups with and without the implementation of optimization

```

// my_total_histories is the number of histories to be simulated from the input file
 unsigned long long int my_total_histories = total_histories;


 size_t Nb = 0; // dynamic shared memory per block

 int blockSize;    // NUMBER OF THREADS PER BLOCK
 int minGridSize; // The minimum grid size (number of blocks) needed to achieve the maximum occupancy for a full device launch
 int gridSize;     // NUMBER OF BLOCKS


 // Returns the grid and block sizes that achieves maximum potential occupancy for the track_particles kernel fucntion.
 cudaOccupancyMaxPotentialBlockSize( &minGridSize, &blockSize,
                                     track_particles, 0, 0);


 int maxActiveBlocks; // Number of active warps for our launch configuration

 // Returns the number of active blocks for track_particles function for a given number of threads per block in a single SM

 cudaOccupancyMaxActiveBlocksPerMultiprocessor( &maxActiveBlocks,
                                                 track_particles, blockSize,
                                                Nb);

 int device;
 cudaDeviceProp props;
 cudaGetDevice(&device);
 cudaGetDeviceProperties(&props, device);

 // Defining the total number of blocks as the number of blocks per SM multiplied by the number of SMs for the GPU device
 gridSize = maxActiveBlocks*props.multiProcessorCount;

 // Calculating the number of active warps per SM
 float ActiveWarps = maxActiveBlocks * blockSize / props.warpSize;

 // Calculating the maximum number of active warps per SM
 float maxActiveWarps = props.maxThreadsPerMultiProcessor / props.warpSize;

 // Computing occupancy as the ratio of the active warps and maximum active warps per SM
 float occupancy = (maxActiveBlocks * blockSize / props.warpSize) /
                   (float)(props.maxThreadsPerMultiProcessor /
                          props.warpSize)*100;

 // Dividing the total number of histories by the total number of threads
 int histSize = (int)(((double)my_total_histories)/((double)(gridSize*blockSize))+0.9990);
```

Figure 29: Screenshot with an extract of the code showing the implementation of the optimization functions.

functions. These setups included different geometries with varying numbers of voxels and photons to be simulated. Additionally, simulations were performed using MPI to enable parallelization across multiple GPUs (specifically, two GPUs in the case of the UPC Argos cluster mentioned in Chapter 6). For the MC-GPU simulations, the initially recommended block size and thread size values of 128 and 100 were used.

Table 24 presents the computational times for the different simulations. It is easy to see that the computational time is reduced when the optimization functions are employed. Furthermore, the execution time is divided by a factor of two when MPI is implemented, effectively utilizing two GPUs instead of one.

Table 24: Execution time for different geometries and number of histories with and without the optimization functions.

Geometry (#voxels)	#histories	Execution time (s)		
		MC-GPU (one GPU)	MCGPU-IR (one GPU)	MCGPI-IR with MPI (two GPUs)
3 x 3 x 3 (voxels)	E9	18	8	9
3 x 3 x 3 (voxels)	E10	178	83	42
3 x 3 x 3 (voxels)	E11	1786	850	420
580 x 260 x 300 (voxels)	E9	46	22	11
580 x 260 x 300 (voxels)	E10	466	222	110

The execution time of MC-GPU simulations is influenced by the values selected by the user for the block and thread sizes, such as the values of 128 and 100 used in our simulations. We conducted simulations with various combinations of these values, and while the simulation times differed, they were consistently lower than the execution time without optimization functions. However, it is possible that for a given GPU type and simulation setup, there might exist an even better combination of block and thread sizes than those calculated by the optimization functions. In such cases, the user would need to manually determine and input these optimal values, which could require additional execution time and expertise in CUDA programming.

The primary objective of this work is to develop a skin dose software tool for radiation dosimetry in medical applications. It is reasonable to assume that users of this software may not possess knowledge of CUDA programming. Therefore, a crucial requirement for a skin dose software tool in the medical field is that the parameter values related to simulation performance should be automatically calculated or predetermined. We believe that the implementation of these optimization functions represents the optimal solution to fulfil this requirement, as it consistently provides improved performance across a wide range of simulation setups.

8.2 Clustering algorithm performance

To evaluate the performance of the PyMCGPU-IR clustering algorithm, organ doses were calculated for various clinical procedures, both with and without the clustering algorithm. Specifically, the calculations involved the mean skin dose (MSD), the peak skin dose (PSD), the mean heart dose (MHD), and the mean lung dose (MLD). The REX anthropomorphic phantom was used for all cases (see section 4.1.4).

Table 25 presents the differences in calculated doses with and without the clustering algorithm, along with the number of simulated events for several procedures. The observed differences were below 3% for all cases, which is acceptable considering the experimental KAP's uncertainty. We also evaluated the clustering performance with other procedures, yielding similar results. Additionally, the execution times and the number of simulated photons are provided, clearly demonstrating the significant reduction in execution times achieved through the implementation of the clustering algorithm.

Table 25: Clustering algorithm performance.

Intervention	Number of events		MSD	PSD	MHD	MLD	Execution time (s)		# histories
	Full	Cluster	diff (%)	diff (%)	diff (%)	diff (%)	Full	Clustered	
San Carlos	42	20	-0.01%	-0.24%	-0.10%	-0.05%	588	280	E9
Sckcen 1	108	11	0.00%	1.98%	0.16%	0.13%	1296	132	E9
Sckcen 2	25	8	-0.42%	2.72%	-1.77%	-1.45%	50	16	E8
Sckcen 3	25	7	0.03%	1.66%	2.70%	0.83%	285	80	E9
Dublin	163	50	-0.05%	0.38%	-0.21%	-0.13%	1793	550	E9
Malmo	186	27	1.31%	2.23%	-0.95%	-2.42%	2418	351	E9

8.3 PMMA measurements

8.3.1 Experimental setup for the PMMA measurements

Measurements were conducted at the Centre Hospitalier Universitaire de Liège in Belgium [53] in a cardiology theater equipped with a Siemens Axiom Artis angiographic system by SCK researchers within the framework of the MEDIRAD project.

Polymethyl methacrylate (PMMA) plates with various thicknesses were irradiated. The PMMA plates were used to simulate patients ranging from pediatric or small adults to large adults [53]. Hereafter, the PMMA irradiations for each PMMA thickness will be referred to as a procedure. Each procedure consisted of 2-3 irradiation events, all performed in a Posterior-Anterior (PA) direction. The setup remained consistent for each procedure and is depicted in Figure 30.

In each procedure, the PMMA dimensions were 30 x 30 x h cm^3, where h represents the thickness and varies for each procedure (h = 10, 15, 20, 25, 30 cm). For each procedure, the PMMA was positioned on a jig made of 3 cm thick extruded polystyrene foam, which included a recess for the TLD detectors. The PMMA, along with the extruded polystyrene foam, was placed on a mattress on a table. The combined thickness of the mattress and table was 7 cm. The positions of the different elements were referenced with respect to the Isocenter (ISO), which is located at 75 cm from the source. Specifically, the positions were indicated as ISO-x, where x represents the distance below the isocenter. The distances from the isocenter are as follows: ISO-18 for the TLD dosimeters, ISO-15 for the bottom surface of the PMMA, and ISO-25 for the table height. These distances from the beam source are illustrated in Figure 30: table at 50 cm, TLDs at 57 cm, and PMMA at 60 cm height. A Radcal detector was placed next to the TLDs to measure the air kerma.

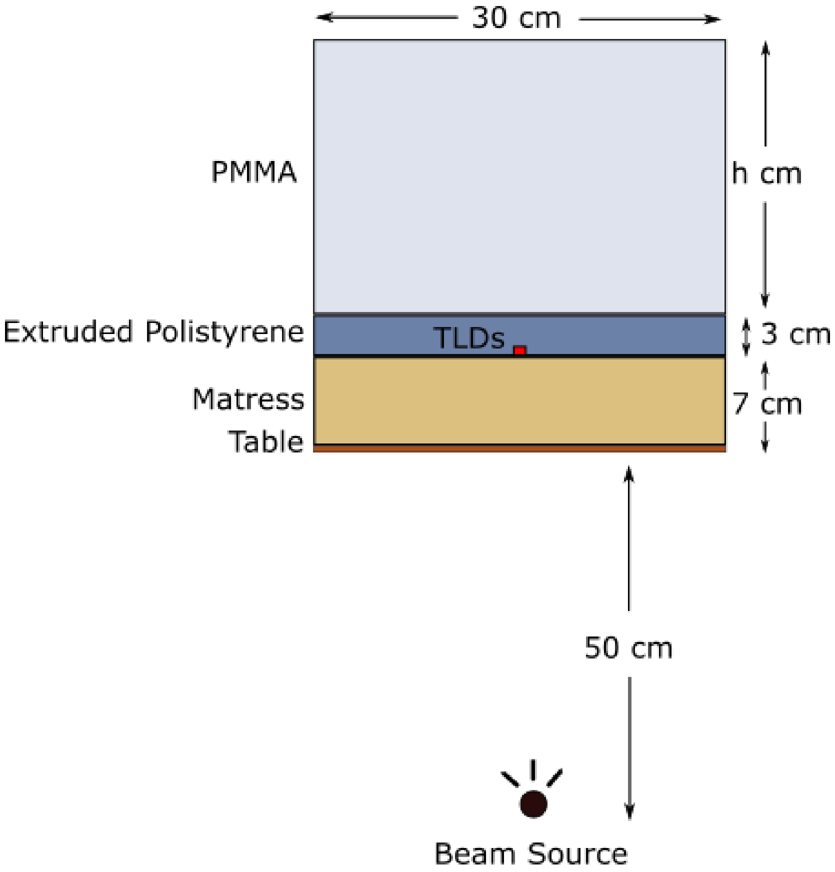


Figure 30: Experimental setup for the PMMA measurements.

Table 26 presents the kVp, field width at 120 cm from the source, and air kerma at the reference point (located at 60 cm from the beam source) for each recorded irradiation in the RDSR, corresponding to each phantom height

Table 26: Parameter values of the experimental setup for the PMMA measurements

PMMA thickness (cm)	Projection	kVp (kV)	Field width (cm)	Dose (RP) (mGy)
10	P1	92	15.77	8.37
	P2	92	15.77	328.39
	P3	88	15.77	174.93
15	P1	77	15.77	2.23
	P2	80	15.77	1.11
	P3	100	15.77	633.43
20	P1	102	15.77	2.64
	P2	108	15.77	678.44
25	P1	125	15.77	0.70
	P2	109	15.77	3.12
	P3	108	15.77	1437.81
30	P1	125	15.77	9.15
	P2	125	15.77	2.61
	P3	125	15.77	3.28
	P4	125	15.77	1714.25

Table 27 displays the TLD measurements. The TLDs were positioned centrally on the mattress in the incident direction.

Table 27: TLD measurements in the PMMA phantom.

PMMA thickness (cm)	TLD (mGy)
30	1692
25	1299
20	567
15	461
10	344

The table and mattress combination cause attenuation of the incident energy photons. To incorporate this attenuation in the simulation geometry, the thickness equivalence of Aluminium was determined experimentally using the method described in section 4.5.

Air kerma ($K_{a,r}$) was measured with and without the table and mattress using a Radcal detector for two different beam energies: 80 kV and 120 kV. A transmission factor of approximately 54% for 80 kV and 69% for 120 kV was measured. This indicates an attenuation of approximately 46% and 31% for 80 kV and 120 kV, respectively.

To replicate the above-mentioned measurements, PyMCGPU-IR was used with different thicknesses of Aluminium. Table 28 presents the results for three different thicknesses: 4 mm, 5 mm, and 6 mm. The closest attenuation percentages to the experimental values for both 80 kV and 120 kV were obtained with a 5 mm thick Aluminium table. Additionally, the attenuation produced by a polyurethane table was investigated. It was found that a polyurethane thickness of 10 cm provided a similar attenuation to that of the 5 mm Aluminium table. Based on this finding and for the sake of simplicity, a 5 mm layer of Aluminium is used to simulate the table and mattress in the subsequent calculations for irradiations performed at the cardiology theater of the Centre Hospitalier Universitaire de Liège.

Table 28: Beam attenuation produced for different beam energies and table thicknesses

Aluminium	80 kV		120 kV		Attenuation (%)	
thickness	no table	table	no table	table	80	120
(cm)	(eV/g/hist)	(eV/g/hist)	(eV/g/hist)	(eV/g/hist)	kV	kV
0.4	0.88	0.52	0.77	0.58	41%	25%
0.5	0.88	0.43	0.77	0.54	51%	30%
0.6	0.88	0.41	0.76	0.47	53%	39%

The TLDs were simulated as a box of air with dimensions of 0.75 x 0.1 x 0.75 cm^3. The experimental configuration employed by PyMCGPU-IR for the simulation is illustrated in Figure 31.

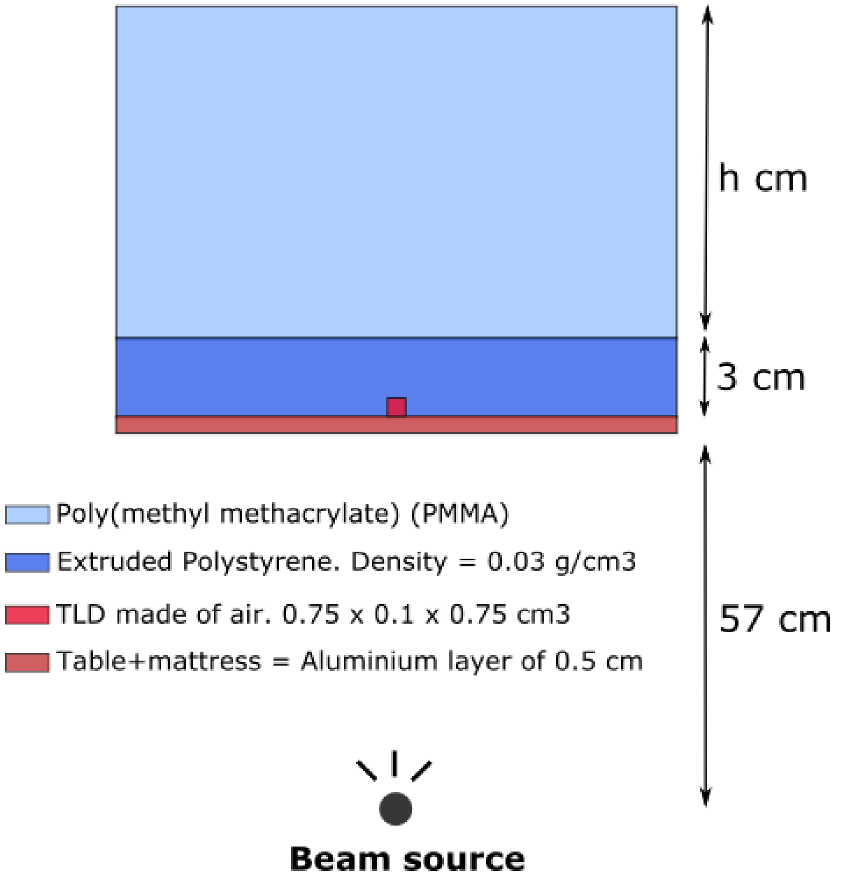


Figure 31: Simulated setup by the PyMCGPU-IR.

8.3.2 Results for the PMMA measurements

Table 29 presents the results for each procedure [78]. Columns 4 and 5 display the total doses in mGy calculated by PyMCGPU-IR for each procedure, along with their associated uncertainties for $k = 2$. The uncertainties account for both the statistical uncertainty of the calculation and a 10% uncertainty associated with the KAP measurement. Columns 2 and 3 show the total doses in mGy experimentally measured for each procedure, accompanied by their associated uncertainties of 20% for $k = 2$. Columns 6 and 7 show the comparisons between PyMCGPU-IR and the experimental measurements for each procedure. The differences are below 30% for all cases. Higher differences are observed for low beam energies, specifically 90 kV and 100 kV, with differences around 30%. For higher beam energies, the differences range from 1% to 10%.

Table 29: Results for the PMMA measurements

PMMA	Experimental		PyMCGPU-IR		Difference	
thickness (cm)	D (mGy)	U(%)	D (mGy)	U(%)	$\frac{\text{PyMCGPU-IR}}{\text{Experimental}}$	U$_{ratio}$ (%)
10	344	20%	444	9%	1.29	22%
15	461	20%	590	13%	1.28	24%
20	567	20%	624	13%	1.10	24%
25	1299	20%	1323	13%	1.02	24%
30	1692	20%	1826	12%	1.08	24%

8.4 Skin dose measurements

8.4.1 Experimental setup for the skin dose measurements

Measurements were conducted at the Centre Hospitalier Universitaire de Liège in Belgium [53] in a cardiology theater equipped with a Siemens Axiom Artis angiographic system as part of the MEDIRAD project. The TLD results were provided by our partners at SCK·CEN.

The purpose of this experiment was to measure the skin dose for different beam angulations using a realistic patient representation. For this purpose, an adult RA anthropomorphic phantom, representing a patient weighing approximately 75 kg, was irradiated in four setups (see Table 30).

The first setup involved a Posterior Anterior (PA) irradiation. The second setup comprised a combination of PA and Lateral Oblique (PALAO) irradiation. The third setup involved a combination of PA, Left Anterior Oblique (LAO), and Cranial (PALAOCRA) irradiations. Finally, the fourth setup used a Lateral (LAT) irradiation.

For the PA, PALAO, and PALAOCRA setups, the beam energy was set to 77 kV, with 3 mm of Aluminium filtration and 0.1 mm of Copper filtration. As for the LAT irradiation, two beam energies of 93 kV and 89 kV were used, both with 3 mm of Aluminium filtration and 0.1 mm of Copper filtration.

The Crano-Caudal and Lateral beam angulations for the PA, LAO, Cranial, and LAT irradiations were as follows: (0^{o}, 0^{o}), (20^{o}, 0^{o}), (0^{o}, 15^{o}), and (90^{o}, 0^{o}), respectively.

Table 30: Parameter values for the experimental setup for the skin dose measurements. CW means clock wise and CCW mean counter clock Wise

Projection	Beam angles		Energy (kV)	Filtration	
	x-y CW	y-z CCW		Copper (mm)	Aluminium (mm)
PA	0.00	0.00	77	0.1	3.0
PALAO	20.00	0.00	77	0.1	3.0
PALAOCRA	0.00	15.00	77	0.1	3.0
LAT	90.00	0.00	93	0.1	3.0
			80	0.1	3.0

The RA phantom was placed on a mattress, and the TLD detectors were positioned on the mattress beneath the phantom, specifically centered on the patient's heart.

For the PA and LAT irradiation setups, three TLDs were placed in a small bag. In contrast, for the PALAO and PALAOCRA irradiation setups, a grid of 25 TLDs was used to ensure the recording of the Position-Sensitive Detector (PSD). The grid consisted of 5 x 5 TLDs with a separation of 3 cm between each detector, covering a total area of 12 x 12 cm^2.

Figures 31, 32, and 33 show the relative positions between the X-ray source and the RA phantom for the PA, PALAO, and LAT irradiation setups, respectively. The height of the table was selected so that the top of the mattress aligns with the patient entrance reference point, which is situated 15 cm below the isocenter. As illustrated in the figures, this arrangement resulted in a source-to-isocenter distance of 75 cm.

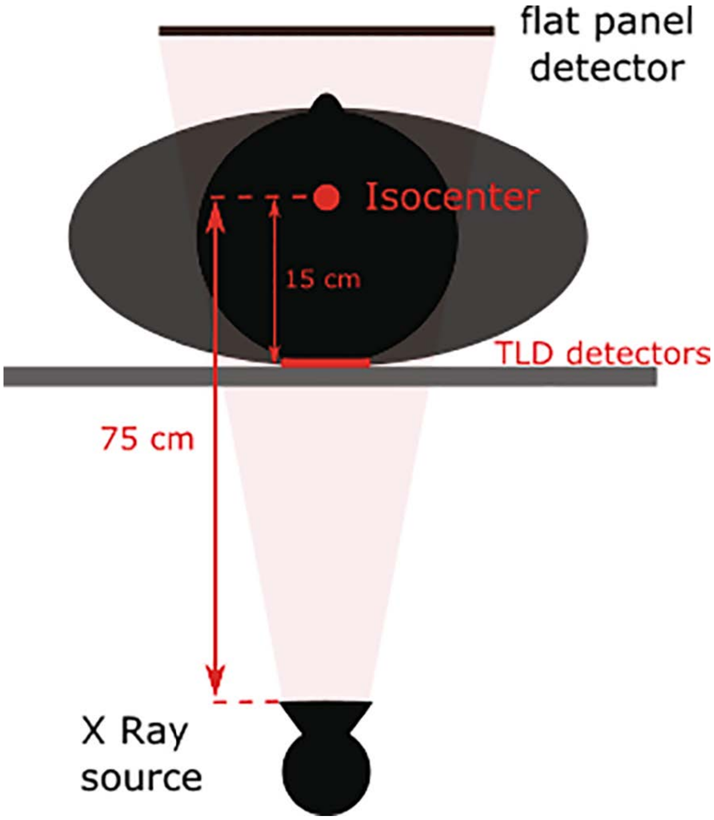


Figure 32: PA irradiations setup.

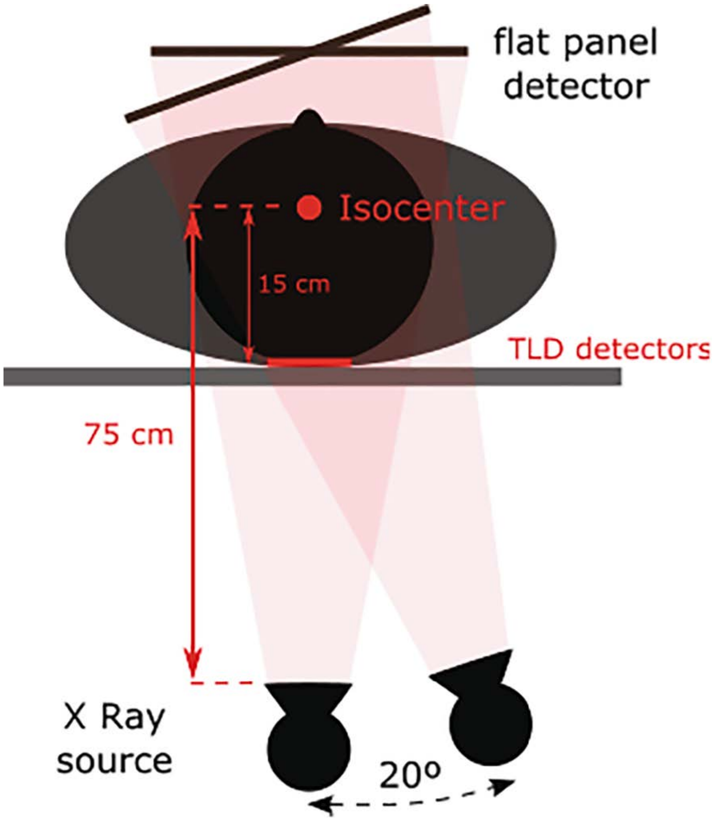


Figure 33: PALAO irradiations setup.

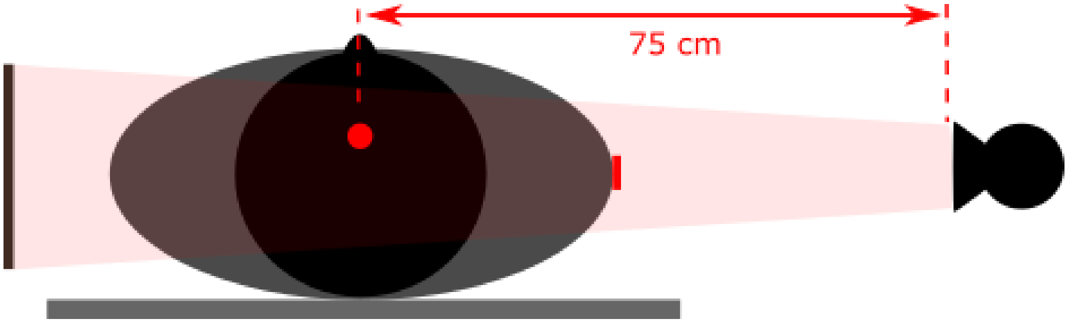


Figure 34: LAT irradiations setup.

PyMCGPU-IR was used to replicate the above-mentioned measurements, with the radiological information obtained from the Radiation Dose Structured Report (RDSR).

The current version of PyMCGPU-IR incorporates the Rex and Regina anthropomorphic phantoms, representing male and female patients, respectively. However, to minimize uncertainty related to the phantom geometry in this validation, the RA phantom was introduced in PyMCGPU-IR. Several simulations were conducted using both the Rex and RA phantoms to analyze the influence of the phantom.

For the skin dose setup, this is not a limitation. The back of the Rex and Regina phantoms is segmented and thus they can be used to calculate the peak skin dose, defined as the maximum dose from a single voxel. Considering that the RA phantom represents a male patient, the Rex phantom was employed for the simulations.

The voxel model of the RA phantom was generated from CT images of the phantom used in the measurements [53]. The voxelized RA phantom consists of 256 x 361 x 185 voxels, with dimensions of 0.15 x 0.1 x 0.5 cm^3. It is segmented into five different materials: air, soft tissue, bone, lung tissue, and air cavities. Small cavities were included in the phantom to accommodate the TLD detectors, each with a volume of 0.11 cm^3, and they were considered to consist of air in the simulations.

In the case of the skin dose setup, the clustering algorithm was not applied. This is because the irradiation involved only one, two, and three differentiated events for the PA, PALAO, and PALAOCRA setups, respectively, and two events for the LAT setup.

8.4.2 Results for the skin dose calculation

Table 31 and Table 32 present the measured and calculated peak skin dose (PSD) along with their associated uncertainties for the first set of measurements described in section 4.8. Table 31 corresponds to the REX phantom, while Table 32 corresponds to the RA phantom [41].

Since there were only three TLDs and they were uniformly irradiated in the PA and LAT setups, the mean value of the TLD doses was taken as the experimental PSD. For the PALAO and PALAOCRA setups, the maximum TLD dose measurement from the array of 25 TLDs was considered as the experimental PSD. It should be noted that, in the case of the REX phantom, the simulated dose with PyMCGPU-IR could not be performed. This is because, unlike the RA phantom, the REX phantom has arms, and thus the beam source irradiates the REX arm instead of the lateral region of the body as it occurs in the experimental setup with the RA phantom.

The differences between the calculated and TLD measurements for the PSD ranged from -6% to +6% for the REX phantom, and from -7% to +7% for the RA phantom. These differences were consistent with the associated uncertainties. As an illustrative example, Figure 35 depicts the simulated PSD and skin dose distribution for the PALAOCRA setup.

The disparities in the PSD between the simulations using both the REX and RA phantoms were within 10% for the three simple irradiation setups.

Table 31: Results for the skin dose experiment for the REX phantom

Projections	TLD measurements		PyMCGPU-IR		Ratio		Exec. time (s)
	PSD (mGy)	U(%) $(k=2)$	PSD (mGy)	U(%) $(k=2)$	$\frac{\text{PyMCGPU-IR}}{\text{TLD}}$	U(%) $(k=2)$	
PA	337.02	20%	355.97	7%	1.06	21%	90
PALAO	511.72	20%	480.76	15%	0.95	25%	180
PALAOCRA	727.49	20%	741.58	12%	1.02	23%	270
LAT	891.00	20%	-	-	-	-	-

Table 32: Results for the skin dose experiment for the RA phantom

Projections	TLD measurements		PyMCGPU-IR		Ratio		Exec. time (s)
	PSD (mGy)	U(%) $(k=2)$	PSD (mGy)	U(%) $(k=2)$	$\frac{\text{PyMCGPU-IR}}{\text{TLD}}$	U(%) $(k=2)$	
PA	337.02	20%	314.81	7%	0.93	21%	100
PALAO	511.72	20%	437.75	15%	0.86	25%	200
PALAOCRA	727.49	20%	679.18	12%	0.93	23%	300
LAT	891.00	20%	897.27	4%	1.01	20%	200

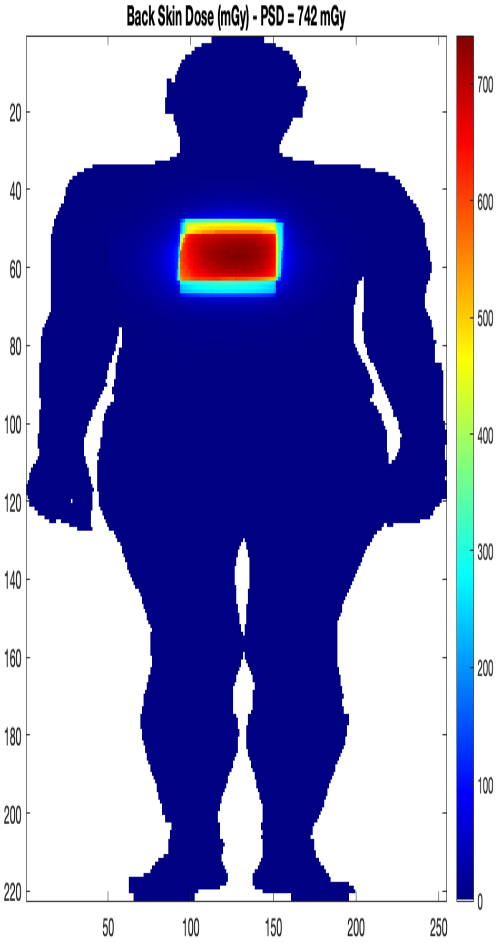


Figure 35: Skin dose distribution for the PALAOCRA (PA + LAO 20º + CRAN 15º) irradiation setup. Voxel size of 0.21 × 0.21 × 0.80 cm^3

8.5 Organ dose measurements

8.5.1 Experimental setup for the organ dose measurements

Measurements were conducted at the Centre Hospitalier Universitaire de Liège in Belgium [53] in a cardiology theater equipped with a Siemens Axiom Artis angiographic system as part of the MEDIRAD project. SCK researchers provided us with the results.

Three clinical procedures were performed, namely, procedures Set1, Set2, and Set3, consisting of 108, 25, and 25 events, respectively. The RA phantom comprises a series of slices, each containing small perforated cavities for inserting TLD detectors. In this experiment, TLD detectors were placed in five different slices in both the right and left lung and at the level of the esophagus. The TLD detectors were placed in slices 14 to 18, which approximately correspond to the region of the heart. In Figure 36, slices 14 to 18 are depicted along with the perforated cavities within them. The cavities with a red circle indicate the locations where TLD detectors were positioned. Additionally, the number associated with each TLD detector is displayed next to the red circle and was used for identifying the TLD detectors to subsequently compare the measured dose with the corresponding simulated dose at the TLD detector level. Gafchromic films (refer to section 5.2) were used during these procedures to assess the skin dose distribution.

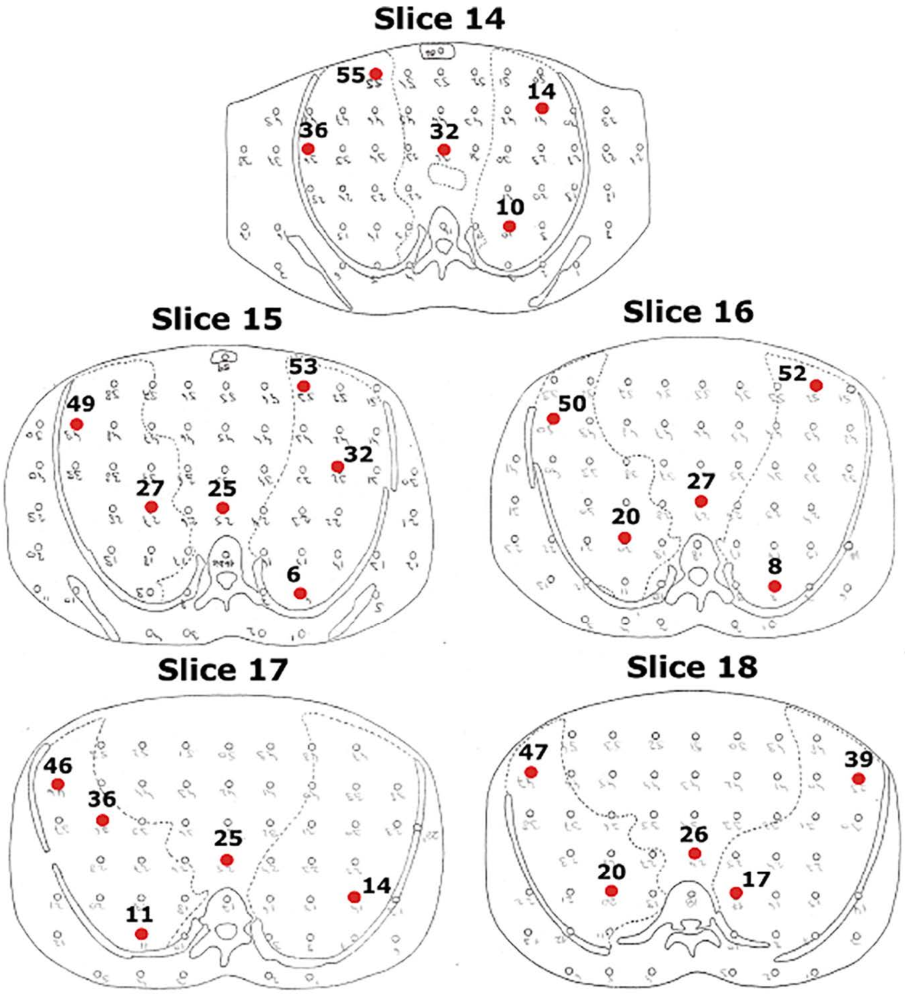


Figure 36: Slices 14 to 18 with the perforated cavities where the TLD detectors are placed. The red circles indicate those cavities where a TLD detector was located together with the associated number for identification purposes.

PyMCGPU-IR was used to replicate the above-mentioned measurements. PyMCGPU-IR relies on the

Radiation Dose Structured Report (RDSR) for dose calculations.

For the organ dose measurement setup, to ensure a comprehensive and accurate comparison, we employed the same voxelized RA phantom used in the skin dose measurement setup described in section8.4.1. Before incorporating the voxelized RA phantom into PyMCGPU-IR, we made modifications to assign different indexes (all consisting of air) to the cavities illustrated in Figure 36. This adjustment enables the automatic calculation of the dose in each cavity using PyMCGPU-IR.

8.5.2 Results for the organ dose calculations

The results obtained from the three clinical procedures are presented in the tables below [41]. Set1 comprised 108 events, which were reduced to 11 after applying the clustering algorithm. The calculation time for Set1 was 15 minutes. Set2 consisted of 25 events, reduced to 8 calculations, with a calculation time of 12 minutes. Set3 involved 25 events reduced to 7 calculations, and the calculation time was 10.5 minutes.

Both the experimental and simulated doses were obtained for each TLD, categorized by organ (left lung, right lung, and esophagus) and per slice. The average dose for both the experimental and simulated doses was calculated as the mean dose of the corresponding TLDs for each slice and organ. In Table 33, Table 34, and Table 35, the measured and calculated absorbed doses are presented as average doses per slice, separately for each organ, along with the associated uncertainties and the ratio between the calculated and measured doses.

Table 33: Comparison of the measured mean dose of the left lung, right lung and the esophagus with the corresponding calculated doses for Set1

Set1		TLD measurements		PyMCGPU-IR		$\frac{\text{Calculation}}{\text{Measurements}}$	
Organ	Slice	dose (mGy)	U(%) ($k = 2$)	dose (mGy)	U(%) ($k = 2$)	Ratio	U (%) ($k = 2$)
Left Lung	14	9	15%	8	55%	0.87	57%
	15	15	15%	13	15%	0.85	22%
	16	76	17%	90	8%	1.20	19%
	17	115	14%	108	6%	0.94	15%
	18	94	17%	91	7%	0.96	18%
	Average	62	9%	62	4%	1.00	69%
Right Lung	14	14	17%	9	30%	0.69	34%
	15	22	16%	48	9%	2.13	18%
	16	27	19%	20	14%	0.76	23%
	17	18	20%	16	24%	0.92	31%
	18	28	19%	27	13%	0.98	23%
	Average	23	8%	24	7%	1.12	11%
Esophagus	14	10	20%	14	22%	1.41	30%
	15	23	20%	35	12%	1.53	23%
	16	51	20%	58	13%	1.14	24%
	17	58	20%	72	12%	1.25	23%
	18	70	20%	74	12%	1.06	23%
	Average	42	10%	51	6%	1.20	12%

Table 34: Comparison of the measured mean dose of the left lung, right lung and the esophagus with the corresponding calculated doses for Set2.

Set2		TLD measurements		PyMCGPU-IR		$\frac{\text{Calculation}}{\text{Measurements}}$	
Organ	Slice	dose (mGy)	U(%) ($k = 2$)	dose (mGy)	U(%) ($k = 2$)	Ratio	U (%) ($k = 2$)
Left Lung	14	18	15%	19	25%	1.07	29%
	15	10	14%	11	14%	1.10	20%
	16	95	15%	95	7%	1.00	17%
	17	137	14%	124	5%	0.90	15
	18	118	16%	95	7%	0.81	17%
	Average	75	8%	69	4%	0.91	9%
Right Lung	14	11	16%	6	36%	0.54	39%
	15	44	15%	51	9%	1.18	17%
	16	18	18%	18	14%	0.99	23%
	17	16	20%	14	24%	0.90	31%
	18	22	18%	37	11%	1.66	22%
	Average	22	8%	25	6%	1.14	10%
Esophagus	14	15	20%	16	26%	1.03	33%
	15	31	20%	35	15%	1.13	25%
	16	51	20%	60	11%	1.19	23%
	17	59	20%	65	12%	1.10	23%
	18	64	20%	71	12%	1.10	23%
	Average	44	10%	49	6%	1.12	11%

Table 35: Comparison of the measured mean dose of the left lung, right lung and the esophagus with the corresponding calculated doses for Set3.

Set3		TLD measurements		PyMCGPU-IR		$\frac{\text{Calculation}}{\text{Measurements}}$	
Organ	Slice	dose (mGy)	U(%) ($k = 2$)	dose (mGy)	U(%) ($k = 2$)	Ratio	U (%) ($k = 2$)
Left Lung	14	26	16%	13	53%	0.49	55%
	15	9	13%	9	16%	0.98	21%
	16	87	16%	100	9%	1.16	19%
	17	107	15%	136	6%	1.28	16%
	18	89	16%	106	8%	1.19	18%
	Average	64	8%	73	4%	1.15	9%
Right Lung	14	10	15%	5	40%	0.48	43%
	15	53	15%	64	11%	1.21	18%
	16	12	17%	13	15%	1.13	23%
	17	-	-	-	-	-	-
	18	17	18%	30	12%	1.80	22%
	Average	18	10%	28.08	7%	1.22	12%
Esophagus	14	12	20%	8	29%	0.66	35%
	15	23	20%	34	16%	1.52	26%
	16	28	20%	61	12%	2.17	24%
	17	30	20%	63	13%	2.09	24%
	18	28	20%	60	13%	2.11	24%
	Average	24	9%	45	7%	1.86	11%

Differences between the calculated and measured average organ doses are within 20% for Set1, Set2, and the left lung in Set3. However, in some specific slices, the differences are higher due to the difficulty in accurately reproducing the phantom's position in the calculations. In Set3, the ratio of the average calculated and measured doses is 1.22 for the right lung and 1.86 for the esophagus. Table 35 shows ratios of approximately a factor of 2 for several slices.

Roser et al. [71] conducted a study to assess the uncertainty in organ-equivalent dose in interventional radiology resulting from marginal displacements of the phantom. They discovered that longitudinal translations of ±25 mm led to differences ranging from 6% to 135% in the most irradiated organs, while the skin dose remained nearly constant, with a maximum difference of 5%.

To analyze the influence of phantom positioning, calculations for Set3 were repeated by shifting the phantom's position 7.5 mm along the x-axis (transversal direction) and 15 mm in the z-axis (longitudinal direction). Table 36 presents the results for Set3 with the newly simulated doses, along with an additional column displaying the differences between the new simulated doses and those in Table 35. The ratio between simulated and experimental doses is calculated with the new configuration. The ratio between simulations and measurements slightly improves for specific slices, with ratios below 1.6 compared to a factor of 2.17 in the original case. Additionally, there is an improvement in the average dose for the esophagus. The differences between the new and original simulated doses for each slice and organ are within 60%, while the differences in average doses for each organ are within 20%. Hence, even minimal displacements can result in significant differences at the TLD level, while the doses at the organ level remain relatively consistent.

Table 36: Comparison of the measured mean dose of the left lung, right lung and the esophagus with the corresponding calculated doses for Set3, for two positions of the phantom.

Set3		TLD measurements	x=0, z=0	x=-7.5 mm, z=15 mm			
Organ	Slice	dose (mGy)	dose (mGy)	dose (mGy)	diff (%)	Ratio	U(%) ($k = 2$)
Left Lung	14	26	13.06	2	61%	0.80	42%
	15	9	8.68	9	9%	1.06	26%
	16	87	100.28	128	28%	1.48	22%
	17	107	136.46	150	10%	1.41	21%
	18	89	105.78	121	15%	1.36	21%
	Average	64	72.85	86	18%	1.35	11%
Right Lung	14	10	4.96	7	46%	0.70	37%
	15	53	64.36	82	28%	1.54	22
	16	12	13.21	12	-17%	0.94	26%
	17	-	-	-	-	-	-
	18	17	29.80	16	-46%	0.98	26%
	Average	23	28.08	29	4%	1.27	15%
Esophagus	14	12	8.13	9	13%	0.74	37%
	15	23	34.21	32	-6%	1.42	25%
	16	28	61.07	39	-36%	1.39	24%
	17	30	63.27	49	-23%	1.61	24%
	18	28	60.02	45	-25%	1.58	25%
	Average	24	45.34	35	-23%	1.43	12%

Figure 37 displays the dose distribution and PSD (Peak Skin Dose) for clinical procedure Set2, alongside the Gafchromic film results for the same procedure. A notable observation is the excellent qualitative agreement between the calculated skin dose distribution and the film results.

For the organ dose set-ups, simulations were done with the RA phantom because we needed to identify the position of the TLDs in the phantom. The calculations were repeated for the three clinical procedures using the REX phantom. In these cases, the differences obtained in the simulated PSD between both phantoms were within 20%.

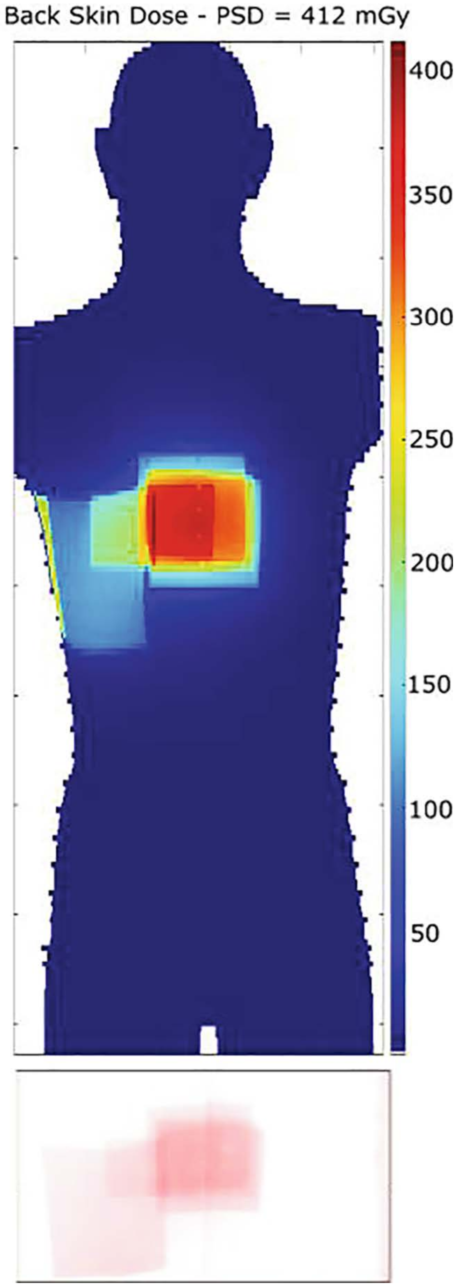


Figure 37: Skin dose distribution for clinical procedure Set2. Figure above, MC simulation (voxel size of 0.15 × 0.1 × 0.5 cm^3); Figure below, Gafchromic film.

Part V

Discussion

Chapter 9

MC-GPU validation

In the comparison of MC-GPU against PENELOPE/penEasy it has been shown that the execution time for PENELOPE running on a single CPU was approximately 2500 times longer than the same simulation implemented on MC-GPU, with differences in the results below 1%. As one of the crucial aspects of this study was comparing the computing times required to complete the simulations using both algorithms, while ensuring agreement between the outputs, the results can be considered highly satisfactory.

The differences between MC-GPU estimates and the doses measured at the UPC Calibration Laboratory were within 5%, but increased to 15% for the hospital setup. However, it's important to note that in all cases, the differences remained within the range of uncertainties. The IEC60580 [22] tolerance for displayed KAP allows for deviations of up to ±35%, while manufacturer's specifications typically allow for ±25% variations. The Spanish quality control protocol in radiodiagnosis establishes a tolerance of 20% for the accuracy of the dose indicator of the X-ray equipment [80], which is consistent with the uncertainty used for the KAP data in this thesis. Moreover, the energy dependence of KAP-meters is often around 10% [81]. Given these factors, the differences observed between the calculations and TLD measurements in this study can be considered excellent, taking into account the inherent uncertainties in this field.

The implementation of the setup at the HUVH hospital using MC-GPU demonstrated execution times of less than one minute. This highlights the advantage of MC-GPU over CPU-based Monte Carlo codes, as it overcomes time limitations and provides a reliable and near real-time tool for dose estimation in clinical practice. However, it is important to ensure that relevant procedure parameters and the KAP value for dose normalization are available for accurate results.

Regarding the MC-GPU comparison against other software performances [82], the results can be considered satisfactory. For example, some studies reported organ dose differences up to 17% when estimations from Dose (QAELUM NV, Leuven, Belgium) were compared against VirtualDoseCT (Virtual Phantoms Inc., Albany, New York, USA). In the specific domain of interventional radiology, VirtualDoseIR (Virtual Phantoms Inc., Albany, New York, USA) organ dose results were compared with measurement data previously reported in literature, with reported differences of 40% [83]. Regarding skin dose estimations, validation studies reported a wider range of differences [24, 34, 30] between 15% and 76%.

As described in section 2.2, most commercial software tools for Skin Dose Calculation (SDC) use a simplified formalism that takes into account factors such as table attenuation and the backscattering of X-ray photons. However, MC-GPU provides the radiation transport of the beam which ensures a closer representation of the patient dose and dose distribution. As demonstrated in Chapter 7, MC-GPU accurately calculates mean doses in tissues, peak doses for different organs, and performs complex analyses like cumulative dose volume histograms. Our results highlight the high performance and accuracy of MC-GPU for dose assessment when compared to standard Monte Carlo codes and TLD measurements. Additionally, it is important to note that MC-GPU not only provides dose values at specific positions, as TLDs do, but also offers information on dose distribution, the position of maximum dose, and organ doses.

Chapter 10

PyMCGPU-IR validation and implementation

10.1 Skin dose calculation with PyMCGPU-IR

The comparison between PyMCGPU-IR and the experimental setups PA, PALAO, and PALAOCRA demonstrates excellent agreement in the calculation of the peak skin dose, with differences below 6%. In a recent article by Krajinovic et al. [84], they presented and validated the SkinCare application (Open Source). As part of the validation process, they conducted tests using a similar setup to measure skin dose and reported that the doses calculated by SkinCare were within 16.9% of the measurements.

Similarly, in the study by Dabin [53], where up to 9 SDC tools were tested against the same skin dose setup on the same angiographic system, the absolute differences were generally larger, ranging within ±40% for all tools and below 20% for 5 of them. Greffier et al. [85] compared the results obtained using the 3D Skin Dose Map tool integrated in DoseWatch®(GE Healthcare) against 98 abdominopelvic embolization or planned coronary angioplasty procedures. The peak skin dose (PSD) was measured using radiochromic films and computed by RDMS (PSDRDMS) using two anthropomorphic phantoms: a flat (2D) phantom and adult male or female anthropomorphic phantoms (3D, ICRP phantoms). The percentage difference between calculated and experimental values for PSD ranged from 35% to 61%.

In another study by Colombo [86], the results of NEXO[DOSE] (BRACCO) in simple phantoms showed that both calculated and measured PSD values agreed with a mean difference of 7% ± 5% (with a maximum difference of 13%).

These results highlight that PyMCGPU-IR outperforms most of the available SDC tools in assessing the peak skin dose.

10.2 Organ dose assessement with PyMCGPU-IR

The comparison between PyMCGPU-IR and organ dose measurements in three clinical procedures is more challenging. The comparison was conducted at the organ level for each slice of the RA Phantom, specifically for the left and right sides of the lung and the esophagus. However, conducting measurements is time-consuming, and there are limitations on the available time for experiments in the hospital. Therefore, the number of TLDs and realistic simulated sets used were limited, and this limitation should be considered when interpreting the results.

For each slice, the dose at the esophagus is measured by only one TLD, while the dose at the left and right sides of the lung is measured by one, two, or three TLDs depending on the slice. In slice 17 of the right lung for Set 3, the TLD measurement was lost during the readout. The differences between calculated and measured doses are within 20% for Set 1, Set 2, and the left lung in Set 3. However, in Set 3, the differences between the calculated and experimental measurements of the dose for the right lung in slice 18 and the esophagus in the five slices are approximately a factor of 2. These differences could be attributed to the uncertainty associated with the positioning of the patient relative to the radiation source. It has been demonstrated that even small displacements in patient position can lead to large differences of up to 60% for the dose at specific points within the phantom. Meanwhile, the overall organ dose shows differences below 20%. Similar behavior has been reported by Roser et al. [71]. Therefore, we believe that the differences observed between the measurement and calculations in the case of Set 3 are not associated with the performance of PyMCGPU-IR but rather with the difficulties in centering the phantom and the fact that there was only one TLD to assess the esophagus dose in each slice. The calibration of the Gafchromic films used was beyond the scope of our work. Therefore, we only used the film images for a qualitative visual comparison with our results.

An important parameter for accurate dose calculation when using Monte Carlo simulation is the phantom used in the geometry input file. It is essential to use a computational phantom that closely resembles the patient anatomy, especially when assessing organ doses. In this study, to minimize the influence of any anatomical differences between the physical phantom used in the clinical procedures and the calculations, a voxel model of the RA phantom was employed for organ dose assessment. However, since PyMCGPU-IR implements the Rex and Regina phantom, the peak skin dose was calculated using the Rex and RA phantom, and it was demonstrated that the results are within 10% for single projections and between 9% and 17% for the clinical procedures. These differences are considered acceptable when compared to the performance of other systems [71]. The results could be further improved by applying a scaling factor to the Rex phantom. Work is in progress to validate and improve the scaling algorithm (see 12.2).

Chapter 11

Limitations of PyMCGPU-IR

11.1 Lack of information in the RDSR

PyMCGPU-IR retrieves the parameter values required to calculate the skin dose from the RDSR (Radiation Dose Structured Report). However, although RDSR is a standard, there may be variations in the representation of certain fields between different machines. Additionally, as described in section 4.7, one of the major challenges for SDC (Skin Dose Calculation) tools is that the system of reference from which the data in the RDSR is derived may be unknown. This is crucial for ensuring an accurate calculation of the skin dose, particularly when determining the location of the peak skin dose. To address these limitations, PyMCGPU-IR includes a complementary file that complements the RDSR. The complementary file can be modified by advanced users or developers to easily adapt PyMCGPU-IR to specific situations. In most cases, this involves conducting prior measurements in the room where the IR (Interventional Radiology) procedure will take place. For example, the table height with respect to the floor needs to be measured and compared with the value provided by the RDSR.

Ideally, one would expect RDSRs from different manufacturers to be harmonized and contain consistent parameter information with standardized definitions. However, significant variations are observed in the reported RDSRs from different manufacturers [87], particularly in technical parameters that are important for skin dose calculation. As mentioned in [87], this emphasizes the need to harmonize both RDSRs and their exports to enable accurate skin dose calculations based on this data. The heterogeneities in the RDSR reports, as well as the requirement for prior measurements to correct the system of reference in the RDSR, introduce limitations as they complicate the automation process of reading and extracting the input parameters needed by PyMCGPU-IR. Currently, PyMCGPU-IR has been proved to make simulations using RDSR from a Siemens Artis Zee, a Siemens Artis Q and a Philips Allura.

11.2 Offline tool

Currently, PyMCGPU-IR is an offline tool, that is, it calculates the dose after the procedure has ended. On the contrary, online tools calculate the dose in real time while the IR procedure is being done. If we would like PyMCGPU-IR to be an online tool, it will need to have access to the live data streaming from the angiographic system. This cannot be done without the permission of the corresponding manufacturer. As explained in the introduction, most of the existing SDC tools are offline systems and extract the necessary data to calculate the PSD from the RDSR. Only the DTS and Dose Map tools are online systems, from CANON and GE Healthcare manufacturer, respectively.

This is a limitation of PyMCGPU-IR in terms of efficiency from the radiological protection point of view. Calculating the skin dose in real time has advantages in comparison of doing it after the whole procedure has finished. For example, if during the procedure the skin dose is being exceeded to some threshold at some region of the patient, PyMCGPU-IR will only alert the doctor after the procedure has finished. This means that the patient may have been irradiated in that same region after the threshold dose was exceeded. If the doctor would have known in real time that the threshold dose was to be exceeded, he probably would have avoided irradiating that region by moving the X-Ray C-arm in order to irradiate from another different angle.

At present, the lack of access to the radiation source information in real time made this a low priority requirement, thus the present version of PyMCGPU-IR does not provide real time dose assessment. However, PyMCGPU-IR could easily be adapted to provide real time results, as soon as the procedure input parameters could be available during the procedure.

The following modifications are necessary to incorporate the feature into PyMCGPU-IR:

The first step in running MCGPU-IR involves loading the patient geometry and materials, and subsequently copying this data to the GPU. This process should ideally be executed prior to the initiation of the patient's irradiation.

Once the program has started, each irradiation event must be converted and sent to MCGPU-IR. The Radiation Dose Structured Report (RDSR) concept isn't applicable here, but it can be interpreted as a series of irradiation events accompanied by metadata. Consequently, PyMCGPU-IR's representation of each event could be used to define the format of the irradiation event transmitted to the server.

11.3 Exact patient location in relation to the table

Having precise knowledge of the patient's exact location is crucial for accurately calculating both the peak skin dose and its location. In some situations, the system reference of the angiographic machine may not be known, thereby leading to uncertainty regarding the position of the table relative to the radiation source. To address this issue, PyMCGPU-IR uses a complementary file that includes the missing information.

However, there is another challenge concerning the patient's location. As described in section 4.7, the coordinates in PyMCGPU-IR have their origin at the lower corner of the patient's left foot. To convert coordinates from the RDSR system to the PyMCGPU-IR system, we make the assumption that the mid-center of the patient intersects the vertical line passing through the isocenter. However, this assumption may not always be accurate. This introduces uncertainties in the skin dose calculation that are not accounted for. One possible solution is to adjust the parameter values in the complementary file, but once again, this would require prior measurements before the IR procedure. Another alternative approach to overcome this problem is to use one of the saved images to improve the alignment between the simulated and experimental geometries.

11.4 PyMCGPU-IR field description

The X-ray beam field simulated by PyMCGPU-IR is pyramidal. While this can be seen as a limitation, it is a common characteristic of commercial C-arms. If the X-ray beam field were not pyramidal, PyMCGPU-IR could attempt to simulate a pyramidal field that closely resembles the given conic field. However, this would introduce uncertainties in the dose calculation. Additionally, certain Philips angiographic machines include wedges in the X-ray source to minimize lung doses. Currently, PyMCGPU-IR does not take this into account.

11.5 Number of C-arms in PyMCGPU-IR

PyMCGPU-IR is designed to simulate irradiations performed by angiographic systems with one C-arm. It is not currently prepared to handle irradiations carried out by bi-plane systems. Implementing support for a second C-arm would require substantial changes to the MC-GPU code, which is beyond the scope of the current thesis. However, this consideration should be taken into account for future versions of PyMCGPU-IR.

Chapter 12

Future work

12.1 Skin dose viewer

For the sake of a better user interface and an easier patient dose interpretation, a tool that automatically reads the retrieved data from PyMCGPU-IR and generates a picture of the skin dose distribution is under progress and will be implemented in new versions of PyMCGPU-IR.

12.2 Geometry scaling

At present, PyMCGPU-IR uses the REX and REGINA phantoms to model a male and a female patient, respectively. The voxelized models of both phantoms have fixed sizes. However, patients undergoing an interventional radiology (IR) procedure can vary significantly in terms of size and shape. To address this, MCGPU-IR offers the capability to scale the voxelized geometry in any direction, allowing the phantoms to be adjusted to better match the actual bodies of the patients. The scaling factor operates by modifying the voxel size in the three dimensions. Hence, the scaling factor consists of three factors, one for each of the three-dimensional dimensions, which alter the voxel sizes accordingly to accommodate the patient's shape and size.

In the results presented in this work, the scaling of the phantoms was deemed irrelevant and, therefore, not applied, as it did not impact our specific cases. However, in the work conducted by [88] within the framework of the PODIUM project [89], the authors employed PyMCGPU-IR and used the scaling factor equations to more accurately replicate the size of the patient phantoms. The parameters used in these scaling factor equations were extracted from the RDSR and consisted of the patient's height and weight. Additionally, the scaling equations in PyMCGPU-IR establish correlations between weight and height values and the dimensions of the patient's body, accounting for both genders and different regions of the body. These correlations were derived from NHANES surveys [90] data and were compared to the Rex and Regina phantoms.

12.3 Uncertainty associated to the patient location

The reliability of the dose calculation is dependent upon the patient's position in relation to the table. Unfortunately, the patient's position on the table is not standardized and is often challenging to document accurately. This issue is commonly encountered in similar software tools as well. As discussed in section 11.3, one potential approach to address this problem is to use one of the images to enhance the alignment between the simulated and experimental geometry or to adjust the parameter values of the complementary file based on prior measurements taken in the clinical room. For future versions of PyMCGPU-IR, it would be necessary to acknowledge and address this problem by developing an approach to overcome it.

12.4 Real time dose calculation

At present, due to the lack of access to the radiation source information in real time, PyMCGPU-IR has been developed to perform calculations once the procedure is finished. However, should this information be available, the programme could be easily adapted as detailed in 11.2. Finally, as regards the clustering algorithm, a cache [1] match could be implemented by using similar rules to compare events [91].

12.5 Integration of patient and operator dose calculation

In this study, we have emphasized that interventional radiology (IR) procedures can expose patients to high levels of radiation. Similarly, although to a lesser extent, medical staff members are also at risk of radiation exposure.

As discussed in subsection 4.3.1, MCGPU-IR was developed within the framework of the MEDIRAD and PODIUM projects [41, 63]. It includes the capability to simulate doses for individual operators and provides the option to simulate the presence of shields placed between the radiation source and the operator. The PODIUM project (Personal Online Dosimetry using Computational Methods) [92, 93] aimed to create a user-friendly online application for calculating radiation doses received by workers during interventional procedures, eliminating the need for physical dosimeters.

---

[1] Cache: an auxiliary memory from which high-speed retrieval is possible.

PyMCGPU-IR uses MCGPU-IR for dose simulations and, consequently, is also capable of simulating operator doses. The feasibility of PyMCGPU-IR for operator dose calculations was tested within the framework of the PODIUM project [94] and was further successfully validated in the PhD thesis of Victor García [88].

The integration of both patient and operator dose calculations in PyMCGPU-IR architecture would considerably enhance radiation protection in the field of interventional radiology. This comprehensive approach sets it apart, making PyMCGPU-IR a more complete solution than any other existing program in the market.

Part VI

Conclusions

Chapter 13

Conclusions

In this thesis, we have developed PyMCGPU-IR, a new software tool based on the Monte Carlo program MC-GPU for assessing skin dose and organ doses in patients undergoing an interventional radiology (IR) procedure. The primary focus of this work has been on validating and improving MC-GPU, as well as creating PyMCGPU-IR, a user-friendly and easily accessible software tool specifically designed for skin dose calculations. PyMCGPU-IR integrates a fast and accurate Monte Carlo code. The main conclusions and milestones of this thesis are as follows:

a) We have developed a new version of MC-GPU, called MCGPU-IR, which incorporates physics models from PENELOPE (2014) and includes several technical tools for automatically setting optimal values for GPU calculations. It has been demonstrated that MCGPU-IR accurately calculates patients' peak skin dose, skin dose distribution, and peak skin dose position in interventional procedures. The differences were below 1% when compared with PENELOPE/penEasy calculations, below 5% when compared with measurements at the UPC Calibration Laboratory, and 15% when compared with TLD measurements in the hospital setup.

b) The calculation time of MCGPU-IR for assessing organ dose in an X-ray event is approximately 2500 times faster than that of standard multipurpose Monte Carlo radiation transport codes (without variance reduction techniques applied).

c) MCGPU-IR has also shown to have a high performance in dose assessment when compared with other software tools.

d) PyMCGPU-IR is an innovative Skin Dose Calculation (SDC) tool that offers higher performance and accuracy for skin dose calculations compared to most available SDCs. Its main features include:

(a) Dose calculations based on Monte Carlo simulation of radiation transport, providing a more accurate description of radiation interactions than calculations based on analytical formulas.

(b) Automatic retrieval of all necessary data for dose calculations from the RDSR (Radiation Dose Structured Report).

(c) Tools for accelerating calculations by grouping sets of events with slight differences in radiological parameter values.

(d) Support for reading a supplement file to introduce manually any relevant information missing in the RDSR.

(e) Successful validation through comparisons with TLD measurements in complex interventional procedures, with differences generally within the associated uncertainty. The differences between the calculated and TLD measurements for the PSD ranged from -6% to +6% for the REX phantom, and from -7% to +7% for the RA phantom. Differences between the calculated and measured average organ doses are within 20%. It should be noted that uncertainties and validations in clinical setups are typically not provided in commercial SDCs.

(f) PyMCGPU-IR provides not only dose values at specific positions, like TLDs, but also dose distribution, position of maximum dose, and organ doses.

(g) Vendor independence of PyMCGPU-IR. Currently, PyMCGPU-IR has been proved to make simulations using RDSR from a Siemens Artis Zee, a Siemens Artis Q and a Philips Allura.

(h) Currently, PyMCGPU-IR provides dose values only after the procedure has finished, as it relies on the RDSR, which is provided afterward. However, with minimal programming effort, PyMCGPU-IR could be adapted to provide real-time dose calculation if real-time radiation source information were available.

List of Figures

List of Tables

Bibliography

[1] Mettler, F. A., Jr *et al.* Radiologic and nuclear medicine studies in the united states and worldwide: frequency, radiation dose, and comparison with other radiation sources–1950-2007. *Radiology* **253**, 520–531 (2009).

[2] Vañó, E. *et al.* ICRP publication 113. education and training in radiological protection for diagnostic and interventional procedures. *Ann. ICRP* **39**, 7–68 (2009).

[3] Balter, S., Hopewell, J. W., Miller, D. L., Wagner, L. K. & Zelefsky, M. J. Fluoroscopically guided interventional procedures: a review of radiation effects on patients' skin and hair. *Radiology* **254**, 326–341 (2010).

[4] Mettler, F. A. Medical effects and risks of exposure to ionising radiation. *J. Radiol. Prot.* **32**, N9–N13 (2012).

[5] Koenig, T. R., Mettler, F. A. & Wagner, L. K. Skin injuries from fluoroscopically guided procedures. *AJR Am. J. Roentgenol.* **177**, 13–20 (2001).

[6] Commission, I. E. Food and drug administration. important information for physicians and other health care professionals. 9 september 1994 (online). `https://webstore.iec.ch/publication/5079` (2005). Accessed: 01.06.2023.

[7] Valentin, J. Avoidance of radiation injuries from medical interventional procedures. *Ann. ICRP* **30**, 7–67 (2000).

[8] ICRP publication 105. radiation protection in medicine. *Ann. ICRP* **37**, 1–63 (2007).

[9] European Society of Radiology (ESR). Summary of the european directive 2013/59/euratom: essentials for health professionals in radiology. *Insights Imaging* **6**, 411–417 (2015).

[10] Stecker, M. S. *et al.* Guidelines for patient radiation dose management. *J. Vasc. Interv. Radiol.* **20**, S263–73 (2009).

[11] ICRP. The 2007 recommendations of the international commission on radiological protection. ICRP publication 103. *Ann. ICRP* **37**, 1–332 (2007).

[12] Jaschke, W., Schmuth, M., Trianni, A. & Bartal, G. Radiation-induced skin injuries to patients: What the interventional radiologist needs to know. *Cardiovasc. Radiol.* **40**, 1131–1140 (2017).

[13] Jaschke, W., Bartal, G., Martin, C. J. & Vano, E. Unintended and accidental exposures, significant dose events and trigger levels in interventional radiology. *Cardiovasc. Radiol.* **43**, 1114–1121 (2020).

[14] MEDIRAD. Implications of Medical Low Dose Radiation Exposure. `http://www.medirad-project.eu/`. Accessed: 01.06.2023.

[15] Cousins, C. *et al.* ICRP PUBLICATION 120: Radiological protection in cardiology. *Ann. ICRP* **42**, 1–125 (2013).

[16] EURADOS. EURADOS Report: Patient maximum skin dose in interventional procedures in radiology and cardiology. 2019. `http://www.medirad-project.eu/`. Accessed: 01.06.2023.

[17] Dabin, J. *et al.* Characterisation of grids of point detectors in maximum skin dose measurement in fluoroscopically-guided interventional procedures. *Phys. Med.* **31**, 1112–1117 (2015).

[18] Chieng, R. *Optically stimulated luminescent dosimeter* (Radiopaedia.org, 2023).

[19] Farah, J. *et al.* Characterization of XR-RV3 GafChromic® films in standard laboratory and in clinical conditions and means to evaluate uncertainties and reduce errors. *Med. Phys.* **42**, 4211–4226 (2015).

[20] Suzuki, S. *et al.* Methods to reduce patients' maximum skin dose during percutaneous coronary intervention for chronic total occlusion. *Catheter. Cardiovasc. Interv.* **71**, 792–798 (2008).

[21] Suzuki, S. *et al.* Patient skin dose during neuroembolization by multiple-point measurement using a radiosensitive indicator. *AJNR Am. J. Neuroradiol.* **29**, 1076–1081 (2008).

[22] Commission, I. E. Medical diagnostic X-ray equipment - Radiation conditions for use in the determination of characteristics. `https://webstore.iec.ch/publication/5079` (2005). Accessed: 01.06.2023.

[23] Federal performance standard for diagnostic x-ray systems and their major components–FDA. final rule. *Fed. Regist.* **59**, 26402–26405 (1994).

[24] Jarvinen, H. *et al.* Feasibility of setting up generic alert levels for maximum skin dose in fluoroscopically guided procedures. *Phys. Med.* **46**, 67–74 (2018).

[25] Malchair, F. *et al.* Review of skin dose calculation software in interventional cardiology. *Phys. Med.* **80**, 75–83 (2020).

[26] Kwon, D., Little, M. P. & Miller, D. L. Reference air kerma and kerma-area product as estimators of peak skin dose for fluoroscopically guided interventions. *Med. Phys.* **38**, 4196–4204 (2011).

[27] Commission, I. E. VERIDIC: Validation and Estimation of Radiation skIn Dose in Interventional Cardiology. `https://www.irb.hr/eng/Divisions/Division-of-Materials-Chemistry/Radiation-Chemistry-and-Dosimetry-Laboratory/Projects/VERIDIC-Validation-and-Estimation-of-Radiation-skIn-Dose-in-Interventional-Cardiology`. Accessed: 01.06.2023.

[28] Malchair, F. *et al.* Review of skin dose calculation software in interventional cardiology. *Phys. Med.* **80**, 75–83 (2020).

[29] Jones AK, P. A. Calculating the peak skin dose resulting from fluoroscopically-guided interventions. part ii: case studies. *Phys. Med.* **13**, 174–86 (2012).

[30] Bordier, C., Klausz, R. & Desponds, L. Accuracy of a dose map method assessed in clinical and anthropomorphic phantom situations using gafchromic films. *Radiat. Prot. Dosimetry* **165**, 244–249 (2015).

[31] Bordier, C., Klausz, R. & Desponds, L. Patient dose map indications on interventional x-ray systems and validation with gafchromic XR-RV3 film. *Radiat. Prot. Dosimetry* **163**, 306–318 (2015).

[32] Nilsson Althén, J. & Sandborg, M. Verification of indicated skin entrance air kerma for cardiac x-ray-guided intervention using gafchromic film. *Radiat. Prot. Dosimetry* **169**, 245–248 (2016).

[33] Daniel R. Bednarek, V. K. R. S. P. N. M. S. J., Jeffery Barbarits & Rudin, S. *HHS Public Access* **118**, 6072–6078 (2016).

[34] Greffier, J. *et al.* Assessment of peak skin dose in interventional cardiology: A comparison between gafchromic film and dosimetric software em.dose. *Phys. Med.* **38**, 16–22 (2017).

[35] Magnier, F. *et al.* Comparison of patient skin dose evaluated using radiochromic film and dose calculation software. *Cardiovasc. Radiol.* **41**, 762–771 (2018).

[36] Hintenlang, D. VERIDIC: Validation and Estimation of Radiation skIn Dose in Interventional Cardiology. `https://virtual.aapm.org/aapm/2018/eposters/217439/david.hintenlang.evaluation.of.fluoroscopic.dose.metrics.predicted.by.dose.html?f=menu%3D14%2Abrowseby%3D8%2Asortby%3D2%2Amedia%3D2%2Aspeaker%3D572428`. Accessed: 01.06.2023.

[37] Habib Geryes, B., Hadid-Beurrier, L., Waryn, M.-J., Jean-Pierre, A. & Farah, J. Benchmarking the DACS-integrated radiation dose monitor® skin dose mapping software using XR-RV3 gafchromic® films. *Med. Phys.* **45**, 4683–4692 (2018).

[38] Benmakhlouf, H., Bouchard, H., Fransson, A. & Andreo, P. Backscatter factors and mass energy-absorption coefficient ratios for diagnostic radiology dosimetry. *Phys. Med. Biol.* **56**, 7179–7204 (2011).

[39] Rana, V. K., Rudin, S. & Bednarek, D. R. Updates in the real-time dose tracking system (DTS) to improve the accuracy in calculating the radiation dose to the patients skin during fluoroscopic procedures. *Proc. SPIE* **8668**, 86683Z (2013).

[40] MakeHuman Community. `http://www.makehuman.org/`. Accessed: 01.06.2023.

[41] Fernández-Bosman, D. *et al.* Validation of organ dose calculations with PyMCGPU-IR in realistic interventional set-ups. *Phys. Med.* **93**, 29–37 (2022).

[42] Badal, A. & Badano, A. Accelerating monte carlo simulations of photon transport in a voxelized geometry using a massively parallel graphics processing unit. *Med. Phys.* **36**, 4878–4880 (2009).

[43] Takata, T. *et al.* Fast skin dose estimation system for interventional radiology. *J. Radiat. Res.* **59**, 233–239 (2018).

[44] Peng, Z. *et al.* A method of rapid quantification of patient-specific organ doses for CT using deep-learning-based multi-organ segmentation and GPU-accelerated monte carlo dose computing. *Med. Phys.* **47**, 2526–2536 (2020).

[45] Frezza, A., Joachim-Paquet, C., Chauvin, M. & Després, P. Validation of irtGPUMCD, a GPU-based monte carlo internal dosimetry framework for radionuclide therapy. *Phys. Med.* **73**, 95–104 (2020).

[46] NVIDIA Corporation. `https://www.nvidia.com/`. Accessed: 01.06.2023.

[47] CUDA architecture. `https://en.wikipedia.org/wiki/Thread_block_%28CUDA_programming%29`. Accessed: 01.06.2023.

[48] C-arm machine. `https://www.medimaging.es/radiografia/articles/294787486/sistema-de-filtracion-optimiza-el-haz-de-rayos-x-en-la-fluoroscopia.html`. Accessed: 01.06.2023.

[49] Digital Imaging and Communications in Medicine (DICOM) Standard, National Electrical Manufacturers Association, Rosslyn, VA, USA. `http://www.dicomstandard.org/`. Accessed: 01.06.2023.

[50] Nuclear Energy Agency. *PENELOPE 2014: A code system for Monte Carlo simulation of electron and photon transport* (OECD, 2015).

[51] IT'IS Foundation. Virtual population. `https://itis.swiss/virtual-population/virtualpopulation/overview/`. Accessed: 01.06.2023.

[52] The Alderson Radiation Therapy Phantom. `https://rsdphantoms.com/product/the-alderson-radiation-therapy-phantom/`. Accessed: 01.06.2023.

[53] Dabin, J. *et al.* Accuracy of skin dose mapping in interventional cardiology: Comparison of 10 software products following a common protocol. *Phys. Med.* **82**, 279–294 (2021).

[54] Zankl, M., Eckerman, K. F. & Bolch, W. E. Voxel-based models representing the male and female ICRP reference adult–the skeleton. *Radiat. Prot. Dosimetry* **127**, 174–186 (2007).

[55] Realistic reference phantoms: An ICRP/ICRU joint effort. *Ann. ICRP* **39**, 3–5 (2009).

[56] Salvat, F., Fernandez-Varea, J. M. & Sempau, J. Penelope2006, a code system for monte-carlo simulation of electron and photon transport (Jul 2008). URL `http://inis.iaea.org/search/search.aspx?orig_q=RN:41016798`. MATHEMATICAL METHODS AND COMPUTING.

[57] Canete, A. M. Optimization of patient dose monitoring in fluoroscopically-guided interventional procedures. `http://hdl.handle.net/2117/117130`. Accessed: 01.06.2023.

[58] Saidi, P., Sadeghi, M. & Tenreiro, C. Variance reduction of monte carlo simulation in nuclear engineering field. In *Theory and Applications of Monte Carlo Simulations* (InTech, 2013).

[59] Gnuplot. `http://www.gnuplot.info/`. Accessed: 01.06.2023.

[60] ImageJ. `https://imagej.net/ij/`. Accessed: 01.06.2023.

[61] GNU Octave. `https://octave.org/`. Accessed: 01.06.2023.

[62] Badal, A., Zafar, F., Dong, H. & Badano, A. A real-time radiation dose monitoring system for patients and staff during interventional fluoroscopy using a GPU-accelerated monte carlo simulator and an automatic 3D localization system based on a depth camera. In Nishikawa, R. M. & Whiting, B. R. (eds.) *Medical Imaging 2013: Physics of Medical Imaging* (SPIE, 2013).

[63] García Balcaza, V. *et al.* Fast monte carlo codes for occupational dosimetry in interventional radiology. *Phys. Med.* **85**, 166–174 (2021).

[64] Nowotny, R. & Höfer, A. Ein programm für die berechnung von diagnostischen röntgenspektren. *Rofo* **142**, 685–689 (1985).

[65] Punnoose, J., Xu, J., Sisniega, A., Zbijewski, W. & Siewerdsen, J. H. Technical note: spektr 3.0-a computational tool for x-ray spectrum modeling and analysis. *Med. Phys.* **43**, 4711 (2016).

[66] Andersson, J. *et al.* Estimation of patient skin dose in fluoroscopy: summary of a joint report by AAPM TG357 and EFOMP. *Med. Phys.* **48**, e671–e696 (2021).

[67] The DICOM standard. `https://www.dicomstandard.org/current`. Accessed: 2023-11-20.

[68] Alves, J. G. *et al.* The new EC technical recommendations for monitoring individuals occupationally exposed to external radiation. *Radiat. Prot. Dosimetry* **144**, 17–25 (2011).

[69] Chiriotti, S. *et al.* Performance of several active personal dosemeters in interventional radiology and cardiology. *Radiat. Meas.* **46**, 1266–1270 (2011).

[70] Parisi, A. *et al.* Photon energy response of LiF:Mg,Ti (MTS) and LiF:Mg,Cu,P (MCP) thermoluminescent detectors: Experimental measurements and microdosimetric modeling. *Radiat. Phys. Chem. Oxf. Engl. 1993* **163**, 67–73 (2019).

[71] Roser, P. *et al.* Pitfalls in interventional x-ray organ dose assessment-combined experimental and computational phantom study: application to prostatic artery embolization. *Int. J. Comput. Assist. Radiol. Surg.* **14**, 1859–1869 (2019).

[72] Kržanović, N. *et al.* Performance testing of dosimeters used in interventional radiology: Results from the VERIDIC project. *Radiat. Meas.* **141**, 106515 (2021).

[73] D. Fernández-Bosman, M. Ginjaume, M.A. Duch, J. Dabin, F. Vanhaere. D2.23 Accelerated Monte Carlo code for an online patient monitoring system in interventional cardiology procedures. `https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5ddf1c904&appId=PPGMS` (June 2021). Accessed: 14.12.2023.

[74] Sempau, J., Badal, A. & Brualla, L. A PENELOPE-based system for the automated monte carlo simulation of clinacs and voxelized geometries-application to far-from-axis fields. *Med. Phys.* **38**, 5887–5895 (2011).

[75] Daures, J., Gouriou, J. & Bordy, J. M. Monte carlo determination of the conversion coefficients Hp(3)/Ka in a right cylinder phantom with 'PENELOPE' code. comparison with 'MCNP' simulations. *Radiat. Prot. Dosimetry* **144**, 37–42 (2011).

[76] Principi, S., Guardiola, C., Duch, M. A. & Ginjaume, M. AIR KERMA TO hp(3) CONVERSION COEFFICIENTS FOR IEC 61267 RQR X-RAY RADIATION QUALITIES: APPLICATION TO DOSE MONITORING OF THE LENS OF THE EYE IN MEDICAL DIAGNOSTICS. *Radiat. Prot. Dosimetry* **170**, 45–48 (2016).

[77] Cross, W., Bomlathm, J., Charles, M., Piesch, E. & Seltzer, S. Dosimetry of external beta radiation for radiological protection (1997).

[78] Fernández Bosman, D. *et al.* Validation of the MC-GPU monte carlo code against the PENELOPE/penEasy code system and benchmarking against experimental conditions for typical radiation qualities and setups in interventional radiology and cardiology. *Phys. Med.* **82**, 64–71 (2021).

[79] White, D. R., Booz, J., Griffith, R. V., Spokas, J. J. & Wilson, I. J. Report 44. *J. ICRU* **os23**, NP–NP (1989).

[80] SOCIEDAD ESPAÑOLA DE FÍSICA MÉDICA, SOCIEDAD ESPAÑOLA DE PROTECCIÓN RADIOLÓGICA, Y SOCIEDAD ESPAÑOLA DE RADIOLOGÍA MÉDICA. *Protocolo Español de Control de Calidad en Radiodiagnóstico, Revisión 2011, SEFM, SEPR y SERAM, Madrid (2011).*

[81] Toroi, P., Komppa, T., Kosunen, A. & Tapiovaara, M. Effects of radiation quality on the calibration of kerma-area product meters in x-ray beams. *Phys. Med. Biol.* **53**, 5207–5221 (2008).

[82] Andersson, J. *et al.* Estimating patient organ dose with computed tomography: A review of present methodology and required DICOM information. a joint report of AAPM task group 246 and the european federation of organizations for medical physics (EFOMP). Tech. Rep., American Association of Physicists in Medicine (2019).

[83] Huo, W. *et al.* VirtualDose-IR: a cloud-based software for reporting organ doses in interventional radiology. *Phys. Med. Biol.* **64**, 095012 (2019).

[84] Krajinović, M., Kržanović, N. & Ciraj-Bjelac, O. Vendor-independent skin dose mapping application for interventional radiology and cardiology. *J. Appl. Clin. Med. Phys.* **22**, 145–157 (2021).

[85] Greffier, J. *et al.* Clinical evaluation of a dose management system-integrated 3D skin dose map by comparison with radiochromic films. *Eur. Radiol.* **30**, 5071–5081 (2020).

[86] Colombo, P. E. *et al.* Validation of a dose tracking software for skin dose map calculation in interventional radiology. *Phys. Med.* **72**, 122–132 (2020).

[87] Dabin, J., Deleu, M., Maccia, C. & Merce, M. S. D 9.141 - standards for digital dose reporting. Tech. Rep., European Joint Programme for the Integration of Radiation Protection Research (2018).

[88] Balcaza, V. G. *et al.* PyMCGPU-IR monte carlo code test for occupational dosimetry. *Radiat. Prot. Dosimetry* **199**, 730–735 (2023).

[89] European Joint Programme for the Integration of Radiation Protection Research. `https://concert-h2020.eu/deliverables/research-projects#anchor-podium%20(2020)`. Accessed: 01.06.2023.

[90] National Center for Health Statistics. `https://www.cdc.gov/nchs/nhanes/index.htm`. Accessed: 01.06.2023.

[91] D. Fernández-Bosman, M. D. U. J. D. F. V. S. C., M. Ginjaume. Accelerated Monte Carlo code for an online patient monitoring system in interventional cardiology procedures . Tech. Rep., Universitat Politècnica de Catalunya (UPC) (2021).

[92] O'Connor, U. *et al.* Feasibility study of computational occupational dosimetry: evaluating a proof-of-concept in an endovascular and interventional cardiology setting. *J. Radiol. Prot.* **42**, 041501 (2022).

[93] Almén, A. *et al.* Personal dosimetry using Monte-Carlo simulations for occupational dose monitoring in interventional radiology: The results of a proof of concept in a clinical setting. *Radiat. Prot. Dosimetry* **195**, 391–398 (2021).

[94] Balcaza, V. G., Camp, A., Sánchez, R. M., Ginjaume, M. & Duch, M. A. Dose assesment with fast monte carlo codes in interventional radiology. *Radiat. Prot. Dosimetry* **199**, 1813–1817 (2023).