UNIVERSITAT DE BARCELONA

Final Degree Project

Biomedical Engineering Degree

"Implementation and automatization of validation tests for the simulation of Nuclear Medicine studies"

Barcelona, 7th June 2023

Author: Arnau Farré Melero

Directors: Pablo Aguiar Fernandez, Jesús Silva-Rodríguez

Tutor: Aida Niñerola Baizán

Abstract

Nowadays, Nuclear Medicine techniques are of great importance in the clinical practice, especially in Neurology and Oncology, for both Diagnostics and Treatment. Focusing on imaging techniques, quantitative methods have shown to be an alternative to the traditional interpreting of images to give support to the clinician for a more accurate diagnostic. The lack of ground-truths has difficulted the implementation of these quantitative methods in the clinical practice, and consequently the use of simulation has raised as a solution for obtaining reference images.

In particular, SimPET is a platform that aims to simulate and reconstruct Positron Emission Tomograph (PET) images while being able to model any scanner by introducing its parameters. The introduction of a new scanner requires a validation process to ensure it works in accordance of the manufacturer's specifications. To do so, the NEMA protocols are a set of tests to allow a comparison with these specifications. In this work we are developing an implementation in Python and automatization of these tests to allow the easy validation of new scanners for SimPET.

Four of the five specified tests were implemented, concerning the Spatial Resolution and the Sensitivity of the tomograph, and the Image Quality and the fraction of Scattering of the reconstructed images.

The results were tested and compared with the experimental ones of the Discovery ST scanner. The results show that the values of Resolution and Sensitivity do not strictly coincide with the experimental ones, and further work on tuning the adjustments of the simulation need to be done. For the Image Quality test we obtained parameters that seem to be in accordance with the theoretical behaviour, but the lack of comparative results does not allow us to extract relevant conclusions. In the case of the Scatter Fraction test, we have shown that the suggested method does not work for very low activities. Thus, due to time and computational limitations further work needs to be done in the development of the techniques, but this work has shown the feasibility of its implementation.

Keywords: Nuclear Medicine, PET, NEMA, Simulation, Reconstruction, Scanner Validation, Monte Carlo.

Acknowledgements

This project would not have been possible without a group of people and their valuable help. First of all, I would like to give my most sincere thanks to Aida, not only for tutorizing this work, but also for all his dedication, effort and interest put into this work, and especially the continuous encouragement I received from her side. Since the first day I felt totally accompanied in this process, and she has given me all the facilities he had at his disposal. It has been an absolute pleasure to work with you during these months and I have learnt a lot from you.

I would also like to show my gratitude to Pablo Aguiar and Jesús Silva, on the one hand for giving me the opportunity to make my contribution on their SimPET project, but on the other hand for their guidelines and orientations on how to overcome some of the difficulties that appeared in the process. Highly helpful have been the recommendations and explanations from Yere and Fran, who I want to acknowledge as well.

Thirdly, I want to thank Roser for disposing all the necessary resources and letting me work in the Biomedical Imaging Laboratory, where has been developed the majority of the project, and ensuring we were comfortable at every moment, but also for helping me choose this topic for my Final Degree Project and all his supportive comments. In the same way, I would not like to forget any of the members of the lab (Raul, Agnès, Arnau, Cris, Alba, Adrià, Raphael, Marc and Sara) with whom I have worked side by side during these months and have made my stay in the lab easier and more enjoyable.

And last but not least, I would like to express a huge thanks to all my fellow biomedical engineers for welcoming me as one more even when I entered this degree later. I appreciate all the good moments and without any doubt, the best part of this experience has been sharing all these years with you.

List of Figures

The following list references the different figures that appear throughout this text.

List of Tables

Following, the list of the different tables present in this memory is presented.

List of Abbreviations

Following are listed the meaning of the abbreviations of some important concepts that will appear in this work.

CPS: Counts Per Second.

FBP: Filtered Back-Projection Reconstruction.

FBP2D: 2-dimensional Filtered Back-Projection Reconstruction.

FBP3D: 3-dimensional Filtered Back-Projection Reconstruction.

FOV: Field of View.

FWHM: Full-Width at Half-Maximum.

FWTM: Full-Width at Tenth-Maximum.

IQ: Image Quality.

LOR: Line of Response.

MC: Monte Carlo techniques.

NEMA: National Electrical Manufacturers Association.

OSEM: Ordered Subset Expectation Maximization.

PET: Positron Emission Tomography.

PHG: Photon History Generator.

QH: Recovery coefficient of the spheres.

RNEC: Noise Equivalent Count Rate.

ROI: Region of Interest.

SENS: Sensitivity.

SF: Scatter Fraction.

SPECT: Single-Photon Emission Computerized Tomography.

SR: Spatial Resolution.

Table of contents

1. Introduction

Nuclear Medicine techniques (particularly Positron Emission Tomography, or PET, and Single Photon Emission Computerized Tomography, or SPECT) are widely used in many clinical scopes such as Neurology. In particular, the use of PET images is crucial in the diagnosis of neurological disorders like epilepsy or dementia. Traditionally these images have been interpreted by visual inspection, but quantification techniques have raised to be an alternative to improve the quality and speed of the diagnostics. [1] These techniques need the obtention of reference images to be used as a ground-truth, which can be obtained by means of simulation [2]. Among the few available alternatives (which we will mention on the *Market Analysis* section) it exists SimPET [3], an open-source project which aims to simulate and reconstruct images from user-created maps. The idea of this program is to be able to model any user-defined PET scan. More about this project and its background will be discussed in the following sections.

1.1. Scope of the project

The main aim of this project is to implement a validation protocol of PET scanner models in order to be used in the SimPET software. SimPET is an open-access platform designed for executing realistic PET simulations. The main structure of the program is already developed and accessible, therefore this project must be seen as a contribution to the source code. Our implementation was based on the NEMA tests, which are protocols provided by the National Electrical Manufacturers Association directed to give an acceptance criterion on whether a scanner is working according to its specifications or not. Particularly, we based on the NEMA-2007 [4] publication, whose details will be expanded on further sections.

This work will be developed in the Biophysics' Image Computing Laboratory facilities, located in the Hospital Clinic, and coordinated with the Pablo Aguiar team in Santiago de Compostela University, who created the SimPET project.

The main limitations of the project are the time and lack of references in the topic. On the one hand, the full implementation of the tests may require a space of time larger than the one expected for a Final Degree Project, so its full integration on the SimPET structure might not be totally completed at the end of the project. On the other hand, there are few resources describing the NEMA tests, which paradoxically require a very faithful implementation to its guidelines. Thus, it is an added difficulty as each step may need to be carefully studied before being translated into code. Furthermore, there are also relatively few references providing scan data to compare with our results. During the development of the project, another limiting factor appeared: the lack of computational resources, especially in terms of free-disk space, that extremely difficulted the handling of results.

Note that this project includes the automatization of these tests until they provide a full set of results, but its integration on a graphical interface or the automatic validation of scanners scape the scope of this work.

1.2. Objectives

According to the scope described above, the main objectives of the project turn around the implementation of the NEMA tests:

- Comprehension and understanding of the NEMA-2007 protocol.
- Implementation of each of the tests.
- Integration of all the tests on a single script.
- Check that their results are consistent and valid for an acceptance tests.
- Depuration of the code in the SimPET architecture.

Deriving from that, other goals may be accomplished depending on the time limitations:

- Optimization of the code.
- Validation of new scanners using the developed code, in order to be ready to be used in the platform.
- Use of one of these scanners for a test study using simulation means.

2. Background

2.1. Clinical background

Nuclear Medicine is the area of Medicine that uses small amounts of radioactive elements mainly for diagnostic purposes, but can also be used in therapeutics and research. Since the discovery of the induced radioactivity in 1934, and its recognition as a medical discipline in 1946 [5] (coinciding with the start of the production of radionuclides with medical purposes), the field has experienced a significant expansion, a tendence that still remains nowadays, particularly in the diagnostics field. [6]

In that sense, many techniques have been developed and integrated in the clinical practice. Some of these techniques include Gammagraphy, SPECT and PET. Particularly, PET (the acronym for Positron Emission Tomography) has gained relevance in Oncology and especially Neuroimaging, while broadening his scope of use in other disciplines like Musculo-skeletal and Cardiology imaging [7]. This technique is also widely used in investigation, e.g., in small animal and bio-distribution studies [8]. In this work we will focus on studies based in PET.

Essentially, this technique is based on the annihilation of positrons with an electron. When this particle-antiparticle pair collide, form two photons of 511 keV energy that travel at opposite directions under the law of conservation of energy. Thus, a positron emitter radiotracer should be chosen (e.g., F-18), and after being injected in the sample or patient, they start colliding with electrons present in the matter. [9]

Hence, the sample is surrounded by a circular scanner, which is in charge to detect these photons. This scanner is formed by a set of parallel rings, which at the same time are constituted by blocks of small scintillation crystals. Note that these are more dense than in conventional gammacameras

as they are set to detect more energetic photons. Very schematically, these blocks include 4 photomultipliers that allow to determine which of the small scintillation crystals has detected the photon. [10] The axial extension of these rings constitute the so-called Field of View, which essentially is the longitudinal distance that the scanner is able to image in one acquisition.

As we introduced above, the annihilation produces two photons, then the scanner is expected to make two detections almost simultaneously for each positron. Thus, the scanner does not directly determine the position of the annihilation, but the so-called Line of Response (LOR), which is the line formed by the detection point of the two photons. Then, what the scanner really stores is the angle and the distance from the centre of the LOR. With these parameters we are uniquely determining the LORs, and the information is saved in the sinograms, which store the number of times a certain LOR is detected. These matrixes are distributed so the x-axis represent the distance from the centre, and the y-axis the angle of the LOR, which are respectively known as offset and projection angle. A sketch of the process can be found in *Figure 1*. As we already commented, the detector is formed by multiple rings: the detection may not be necessarily detected on the same rings, as the sample will be in general tridimensional. Thus, each sinogram will store the information of one pair of rings, so at the end we are going to obtain a matrix of sinograms known as Michelogram (i.e., the element (i, j) corresponds to the detections between the ring i and the ring j)). This transformation from coordinate positions to the projection-offset distribution is known as Radon transform.

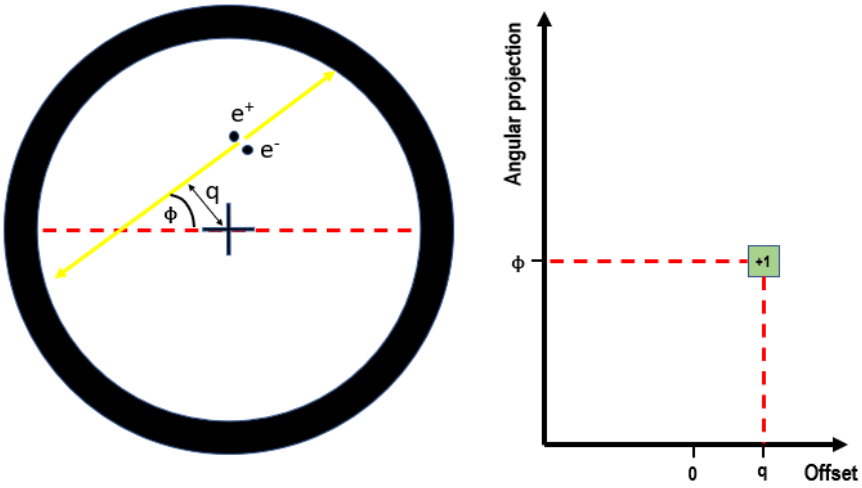

Figure 1. *Sketch of the detection of one photon pair (left) and its storage in the sinogram (right).*

From these sinograms, the final step consists of reconstructing them into the final image. This step is considered as the inverse problem, where from a set of projections contained in the sinogram we may find the tridimensional image of the sample. [11] There are many stablished algorithms widely used in clinical practice in research, which can be generally classified in iterative (e.g., Algebraic Reconstruction Technique) and non-interative methods (e.g., Central Section Theorem). [12] There are even methods based on Deep Learning. However, the reconstruction will not be a central topic of this work, so we will limit ourselves on the following two techniques:

- Filtered Back-Projection (FBP): It is a non-iterative method based on the Fourier Transform of each of the projections p_θ to apply a ramp filter (that is, high-pass) and after recomputing the inverse Fourier Transform, apply the following retroprojection formula to the filtered projections p'_θ:

  $$f(x,y) = \int_0^\pi p'_\theta(x \cdot cos\theta + y \cdot sin\theta)d\theta$$

  The main problematic of this algorithm is the introduction of noise due to the application of the ramp filter, but it has lower computational cost compared to the iterative methods.

- Ordered Subsets Expectation Maximization (OSEM): It is an iterative method which is based on the Expectation Maximization, a metric to measure the likelihood between estimations. In this case, the set of projections is divided into a group of subsets, and for each one, the maximum likelihood is inspected. Thus, several iterations are performed to converge to the Maximum Likelihood solution. In contraposition of FBP, this method has a high computational cost, but it is less sensitive to noise.

In both cases, the acquisitions can be done in 2D or 3D mode: while in the tridimensional case the pairs of all rings are allowed (i.e., we are considering the whole Michelogram) in 2D we are restricting the acquisition to a maximum difference of rings, discarding the farther pairs (purely speaking, the 2D acquisition would accept only the detections on the same ring). [13]

There are many undesired effects that can degrade the quality of a PET image: presence of noise, higher attenuation effect that in the deeper parts of the tissues, the range of displacement of the positron until it reaches an electron, the non-collinearity of the emitted photons… which can produce a reduction of the resolution or the introduction of artifacts. Some of these can be mitigated with some techniques, for example, with the incorporation of a Magnetic Resonance or a Computed Tomograph to the PET scanner.

In particular, in this work we will deal especially with two of these unidealities: the scattering and the random coincidences. The scattering refers to the phenomena where a photon can deviate from its original trajectory when interacting with matter; thus, if there is a certain degree of deviation, the scanner will determine a wrong LOR (as the scattered photon will not arrive to the expected position). The random coincidence refers at the possibility of multiple annihilations at a very similar time. When this happens, we can simply discard the coincidences if we detect more than 2 photons in a very small-time window. But when randomly only 2 photons from different annihilations are detected, the scanner will assume these come from the same pair and build extract its corresponding LOR, leading to wrong information in the sinogram (which is translated into noise in the final image). An sketch of these phenomena is presented in *Figure 2*.

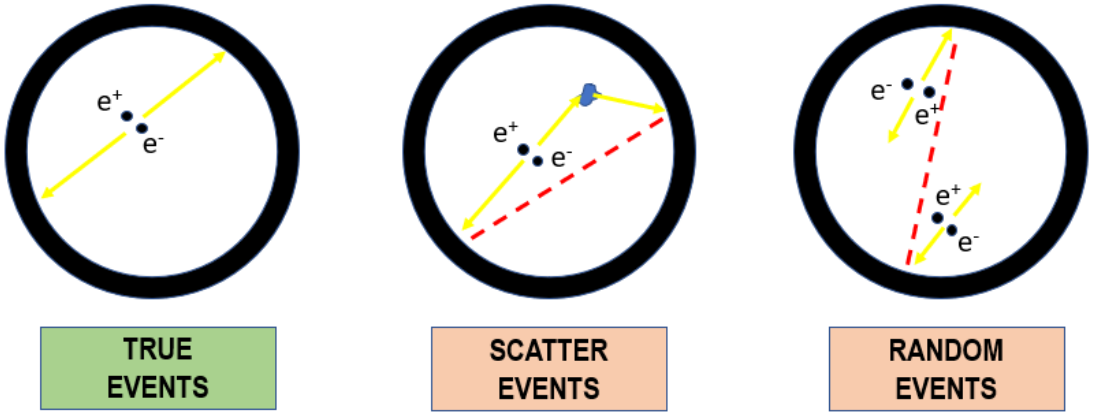

Figure 2. *Sketch of the true, scatter and random events.*

The well-defined LORs, the ones coming from scattered photons and the ones due to random coincidences are known as true, scatter and random events, respectively.

2.2. Relevance of simulation studies in Nuclear Medicine

Traditionally, the medical images (particularly PET imaging) have been interpreted by visual inspection by the clinicians in order to stablish a diagnostic. Although this methodology is still in use in the clinical practice and gives an acceptable performance, the introduction of quantitative methods for the analysis of medical image has brought an improved accuracy and precision on the diagnostic. [14] Besides, it leads to a faster interpretation of the images. These quantitative methods are a set of computational techniques that allow to extract data and information that may be relevant for the doctor and might apport the same information as by manual annotation.

However, quantitative methods need a ground-truth in order to extract results, that is, a reference set of images to compare with. The straightforward approach is the use of phantoms in real scanners that should resemble as better as possible the human geometries and characteristics. A phantom is defined as a specific object used in medical image to evaluate and study the performance of a scanner. [3] These phantoms can therefore be used to obtain reference images, as we perfectly know the object we are imaging. However, these phantoms are difficult to tune and give a small room of manoeuvring, as it is difficult to build a phantom with a very specific geometry, especially a complex one as the human case is. Thus, phantom imaging is said to usually give unrealistic and inaccurate images, which is a very undesirable feature for a reference image.

Hence, the simulation techniques have emerged as an alternative to real phantom imaging. The main idea is to computationally design the phantom as a matrix map and directly simulate the results. With programming tools the possibilities are limited only to the capacity of the user, as there are almost no restrictions on the design of the phantom. Consequently, a large range of clinical subjects and conditions could be represented through simulation. This can have applications on all the fields where PET is well-implemented, especially in neurology where the structures are usually very complex. Digital phantoms can offer a more personalistic solution to represent neurological diseases and disorders. Furthermore, the implementation and deployment of simulation in

detriment of phantom scanning may bring less clinical charge (as now it would not be required to use real equipment to retrieve reference images) and naturally it represents a cheaper solution.

2.3. State of the art: the SimPET platform

The main aim of this project is to implement a validation protocol of PET scanner models in order to be used in SimPET, which is a platform developed on 2021 by the team lead by Pablo Aguiar in the Santiago de Compostela University. [15] It is an open-source project, accessible at: https://github.com/txusser/simpet. This platform aims to bring a simple tool for the simulation and reconstruction of digital phantoms. Ideally, an automatic generation of these phantoms might be implemented, but at the moment it is only possible to input user defined files. SimPET is actually a combination of two programs, SimSET and STIR, that respectively work on the simulation and reconstruction. The objective of this program is to let any user determine the characteristics of the scanner is going to use for the simulation, so the program has to be able to model any kind of scan by knowing its parameters.

SimSET is a program developed by the Washington University Imaging Research Laboratory (1993) which makes use of Monte Carlo (MC) techniques to model physical processes like in a wide range of scopes like Nuclear Medicine. Generally speaking, Monte Carlo refers to a family of techniques that use random sampling, something that is useful in complex systems (e.g., fluids or cellular populations) but also in highly uncertain scenarios. In particular, MC techniques help to capture the sophistication of models like in the photon-matter interaction. Thus, these techniques play with the generation of random numbers to decide the type of interaction or the phenomena involving the photon (e.g., scatter, absortion…). Note that we are working on deterministic phenomena by using randomness. Thus, MC only work on the limit of highly repetitive sampling, otherwise they will not be able to capture correctly the nature of the phenomena simulated.

In *Figure 3* we have simplified the workflow of SimSET. As can be seen, the program is clearly divided in several modules. The core of the program is the Photon History Generator (PHG), which deals with the generation and travel of the photons through the matter. As in real life scanners, the photon (contained in a digital "object") subsequently passes through the Collimator and Detector modules (which do similar tasks as its real homologues) to arrive at the Binning module, which is in charge to deal with the simulated data and convert it to a histogram-like data. In our case, the output that the binning module gives is a set of sinograms. Previous to the PHG, there is an Object Editor module that converts the introduced phantoms by the user to an understandable format for the program (e.g., it specifies what do the numbers in the maps mean). The other modules' behaviour can be tuned with the introduction of a wide range of parameters. More on the parameters and the phantom building will be discussed in the *Concept Engineering* section. Additional files are created to record the decays of the generated photons, which feeds the different modules. Until this point, the workflow is similar for both PET and SPECT simulations, but the PET simulator incorporates an additional module which directly adds random events to the simulated data.

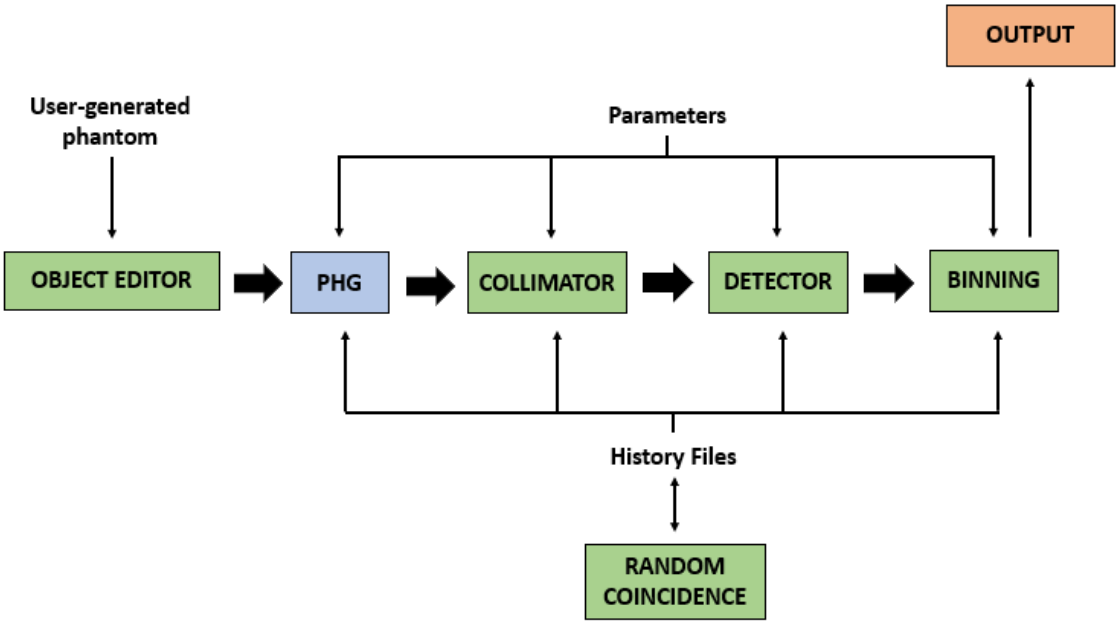

***Figure 3**. Diagram of the workflow of SimSET.*

On the other hand, STIR (Software for Tomographic Image Reconstruction) is in charge of the reconstruction of the generated data by SimSET. Basically, it gathers many algorithms (including the already presented FBP and OSEM) and implement them by taking as an input the generated sinograms. As an output, thus, it brings a reconstructed image.

Hence, SimPET merges the idea of both programs and, by adapting the necessary parameters to a PET environment, this new platform was born. However, it is still in development, as there is a lot of work concerning the tuning of the parameters, the implementation of addition algorithms or the incorporation of new scanners. Precisely this last task is one of the crucial issues in the development of the project, as there is a need to validate them in an efficient and automated way. This work is going to deal more about this feature.

Just for reference, another way of development is the deployment of a graphic interface on the cloud that allow to use the program to the general user population, as currently there is no solid interface and the program has to be controlled directly through scripting and terminal launching. The aim of the website would be to allow to charge or generate the maps, simulate them and reconstruct them, and obtain the intermediate and final results. The current overview of the website (accessible at http://simpetweb.uma.es:8020/) is attached in *Figure 4.*

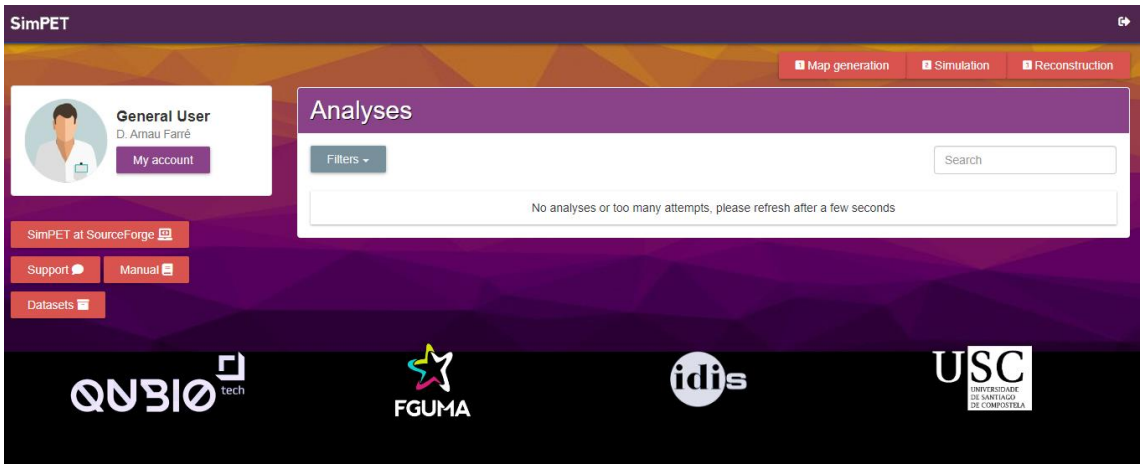

Figure 4. *Current overview of the SimPET clout interface.*

2.4. The NEMA tests

In order to validate the performance of a scanner for its clinical practice, it is required some kind of protocol that allow the technicians to ensure the results are reliable and expected according to the manufacturer's specifications. In that sense, the National Electrical Manufacturers Association (NEMA) periodically publish a set of standards that allow to retrieve measurements on the PET Tomographs. [16]

Although between different publications there may be some changes, the main guidelines have been kept through years. Hence, there are 5 main tests that need to be performed, concerning the Spatial Resolution, the Sensitivity, the Quality of the Image, the Accuracy and the Scatter/Random Counts Measurements. These tests are widely implemented in the hospitals and health centres before bringing these tomographs to clinical practice. As their purpose is, in few words, to check if a scanner is working correctly as expected, it is fundamental that these tests are performed on the simulation models in order to validate them as well. Inconsistent results between the literature and the tests may suggest an incorrect implementation of the simulation model or the need of tuning some of their parameters.

The NEMA publications give details on how the tests should be carried out and impose several restrictions. Therefore, in order to obtain valid results these guidelines must be followed reliably. As a common practice in real clinical environments, each team may dispose of a well-stablished protocol to carry out the tests. However, due to the lack of simulation platforms, there is also a significant lack of implementation of these tests in a programming environment. Furthermore, the execution of these may be inefficient, laborious and cumbersome. Thus, as these processes might be repeated many times (at least once for each scanner) it is strongly recommended to introduce an automatization of these to be executed sequentially. The satisfaction of these needs are indeed the core of this project. In the following table we are summarizing the main information of each of the tests [4] (whose implementation will be discussed in subsequent sections):

Table 1. Summary of the NEMA tests (2007).

Test	Definition	Source distribution	Procedure	Report
Spatial Resolution	Smallest distance distinguishable between two points.	6 small drops of 18-F contained in capillaries.	Acquisition of the 6 source points at the same time and analysis of their profiles, without filtering (most optimistic scenario).	Axial, tangential and radial resolutions.
Sensitivity	Capacity of the tomograph to detect true event photons.	Line source (700 mm) of 18-F	Acquisition of the line source by adding concentric aluminium sleeves and extrapolate the non-attenuated value. Repetition at 10 cm offset.	Total sensitivity of the sensitivity, and sensitivity profile for the first sleeve.
Image Quality	Comparative between the simulated and real images of a certain condition.	4 spheres filled with 18-F and 2 without, surrounded by a radioactive background.	The torso is centred in the FOV of the scanner and imaged during a proportional part to a whole-body 60 minute study. The parameters are the same as in a real study.	Contrast coefficients, background variability and lung insert residual error.
Scatter Fraction	Computation of the fraction of scattered photons in respect of the total.	Line source (700 mm) of 18-F.	The phantom cylinder is centred in the FOV and is periodically imaged during some minutes. The total duration of the scan could reach 14h.	Activity at which the total true count rate events are maximum, and the scatter fraction for this activity.
Accuracy	Ability of the tomograph to correct undesired effects.	Line source (700 mm) of 18-F.	Similar protocol as the Scatter Fraction test, with same source distribution.	Relative error between count rates in different slices.

As mentioned, there may be other tests described in some but not all the protocols, as could be the ones directed for specific tomographs (e.g., the ones equipped with time-of-flight technology). However, these are the more representative and are sufficient for the validation of scanners. In any case, each of the tests uses a standardized phantom with a restricted activity, and require the computation and specification of a set of parameters that shall be reported at the end of the simulations.

3. Market analysis

Following we will briefly analyse to who might be useful the results of this project (and the use of SimPET, consequently) and which alternatives are actually available.

3.1. Potential users

The potential users of the implementation presented in this project (and consequently of the SimPET platform) would be, when all their functionalities get deployed, any clinician using quantitative methods on their PET image analysis. Thus, it could be a professional using one of the already implemented scanners, but also someone that wants to introduce its custom scanner. Of

course, the scope of use of this program extends also to imaging technicians, radiophysicists, and in fact, to any clinical professional that has to deal with PET scans or images.

Of course, the use of SimPET might be extended to companies and research as well that want to perform any kind of simulation through Monte Carlo techniques. In fact, until the program receives the correspondent authorization, it must not be used for clinical purposes and it would be limited to research.

3.2. Alternative platforms

Simulation in Nuclear Medicine has started raising the interest of clinicians and researchers since the last years, being therefore a relatively young discipline. For this reason, it is difficult to find well-stablished alternatives to the proposal of SimPET, even for any typology of imaging. In that sense, the development of SimSET in 1993 (the program that SimPET uses to simulate data) represented a revolution on the simulation field and a starting point of a discipline that has not stopped growing since then.

Some examples on isolated PET simulations can be found in the literature, that is, simulations designed in single codes for an specific scanner and study. Thus, they are not reusable for other applications. For example, in *Assié K. et al (2004)* [17] they perform a study on the validity of the tool-box simulation GATE. *Guérin B. (2008)* [18] brought a comparative study on the use of SimSET, GATE and real data. In *Salvadori J. et al (2020)* [19] they built a specific model for the Philips Vereos scanner and validated its data following the NEMA guidelines. Among the considered papers, this last one is the most closer to our proposal, as we are going to follow the NEMA directrices as well. However, their study was limited and particularized to only one scanner, while we are intending to generalise the use of the NEMA (and in fact, the simulation of PET) to any type of scanner, presenting it as a program and not only as a single script.

In that sense, there are only few examples of programs that follow our similar idea of implementing a workflow of simulation and reconstruction. For example we can find SMART [20] (acronym of SiMulAtion and ReconsTruction), which is a tool able to generate 3D PET images requiring only the introduction of an attenuation map from a Computerized Tomography scan. However, its validity is restricted as they only use the Image Quality test from the NEMA to compare their results.

Another alternative is PeneloPET, a toolkit based on the simulator PENELOPE [21] (analogous to our use of SimSET in SimPET). As the authors describe, it is robust, fast, accurate and can achieve high levels of complexity, but its interface can be complex and unfriendly, as it is based on FORTRAN 77. Hence, it is a very complete alternative and his aims are similar to ours, but as developed in 2006 it is a bit outdated.

Just for reference, we also found an open-source tool developed by *Ashrafinia S. et al. (2017)* [22], based in Matlab that allows analytic simulations of PET (with the possibility to add noise for more realism). Again, their principles seem to relatively match our objectives, but the tool seems to be still in initial phases of development and besides it cannot be found which method of validation they are using.

To sum up, there are not many more examples on PET simulators, at least that are open published and accessible. Consequently, this fact must be taken as an opportunity for our SimPET project to develop something new and that can fulfil a necessity that is still not covered.

3.3. Future perspectives

Thus, taking into account that the use of quantitative methods in PET (and generally in any type of imaging modality) tends to increase with the years, the amount of potential users of SimPET is expected to grow steadily in the following years. Consequently, we are also expecting that the number of studies, tool-boxes and even programs focused on the simulation and reconstruction of Nuclear Medicine image will also grow to fill the growing needs that the clinical world will demand.

4. Concept Engineering

Following we are describing the different aspects that were necessary to study, plan and prepare before starting the actual simulations and implementing the specifications of the NEMA tests.

4.1. Resources and materials

One of the crucial steps in the planning of a programming project, especially one that involves simulations, is the correct election of the computational resources that are going to be used. It is precise to have a computer at our disposal that fulfils mainly three basic requisites:

(i) Enough free disk-space. As dealing with images, the outputs of the programs are expected to be relatively large, especially considering that all the tests have to be save in parallel. Most crucially, the intermediate files (that is, temporal files generated during the pre-processing, post-processing or the simulations themselves that are deleted when the program ends) generated are also expected to include a huge volume of information, depending on the complexity of the case simulated. Consequently, it is not enough to ensure space for the final images, but also a space margin has to be secured, otherwise the simulation will not be able to end and an error might be prompted. According to the developers of SimPET, these files may reach 2 TB in extreme cases.

(ii) High RAM. As higher the RAM of a computer is, the larger its efficiency on executing the tasks. This is significantly important in the simulation scopes, as many instances are launched, and multitasking is very common in these kind of developments. Consequently, it is desirable to work in a computer with a RAM larger than 8GB.

(iii) Multiple Cores. The RAM plays an important role on speeding up the processes, but another way to speed them significantly is by using multiple cores, that is, subdividing the simulation and executing them in parallel. In other words, by launching a simulation in N cores, we expect it to be approximately N times faster. Ideally would be acceptable to work with 6 or more cores.

In that sense, the first option was to use the personal laptop (Windows 10 Intel Core i-7, 2 cores, 8 GB RAM, 750 GB Disk-Space) as the main tool for obtaining the results. This was actually rapidly discarded after doing a few test simulations, as the time needed to complete them was excessive, even for low-dose and short ones.

Besides, it has to be taken into account that SimSET (and therefore, SimPET) exclusively works on Linux Systems. [23] Thus, if the selected computer does not have Ubuntu or another Linux OS installed, additional steps would need to be done. Between the most recommendable options we find installing a Virtual Machine (e.g., VirtualBox), which may bring an uncomfortable interface to work with (especially for a long-term project), or a Linux Subsystem for Windows (WSL), which in our particular case was not possible due to incompatibilities.

For all these reasons, we opted for using the Mercury Computer, ubicated in the Lab Imaging Systems in Hospital Clínic. On the one hand, it already had Ubuntu installed and all the pre-requirements needed to start using SimPET. On the other hand, it seemed to be powerful enough to perform simulations at SimPET level and fulfilled all the requirement we described before. Besides, it made more accessible the resources on the Nuclear Medicine Department, which could be occasionally used.

Although we already mentioned that SimPET has a graphical interface on the cloud, it is not fully functional and practical enough to perform this project. Hence, we will use the standard way by launching Python scripts directly from terminal. This is why we did not had actually a range of options to choose in terms of programming languages. Nevertheless, this fact is not really a problem, as Python is a widely used high-level language in the scientific scope (and in the university as well) and seems to perfectly suit the level of coding of NEMA. Especially practical are some associated libraries like *skimage*, *sklearn*, *nibabel* or *PIL* for the handling and processing of images, something that will be of great importance. In order to test the code and for graphical purposes, the application Jupyter Notebook will be mainly used as it allows to easily modify the code while visualising the results (in front of other APIs like Spyder). In any case, the final code has to be contained in one or multple '.py' scripts.

4.2. Methodology

As mentioned, this project can be understood as an addition to a bigger project which is still in development, SimPET. For this reason, it is really important that all the participants coordinate their actions in order to not overlap work and to avoid incompatibilities between updates of the project. Consequently, due to the high dimensions of the SimPET project, a direct transfer of code and information cannot be considered.

Instead, the use of GitHub repositories has been preferred, especially as SimPET was already conceived as a repository. It allows the opening of branches, which are partial or full copies of the repositories that become independent of the main branch and allow to work with it without disturbing the development of another programmers.

There are many types of workflows available in GitHub, but as accorded with the other members of the group, we will base our work in the GitFlow methodology [24] which is characterised by the

stablishment of a master branch (i.e., the “official version” of the program), a development branch (i.e., the next changes that are going to be applied) and feature branches. These last ones are the represent small changes done on the development branch (as the feature and master branches never interact directly), and when enough changes are done, the union between the development and master branches is performed, the so-called merging, which not only adds the new features to the main program, but also ensures that everything is working correctly. In opposition to the trunk-based workflow, here the development branch is more long-lived and less mergings are required, something that suits better the project rather than continuously performing small changes. There would not be any advantage and besides, it would be very time-consuming. A very simplified scheme of how a GitFlow project may look like can be found in *Figure 5*.

In consequence, ideally the final result of this project may be one single script implementing all the tests, with adding the minimum files as possible. That should allow a easier and simpler merging, without many noticeable modifications in the truncal part of the directories. In principle, no modifications should be done in the code itself unless an important problematic is found during the development of this project.

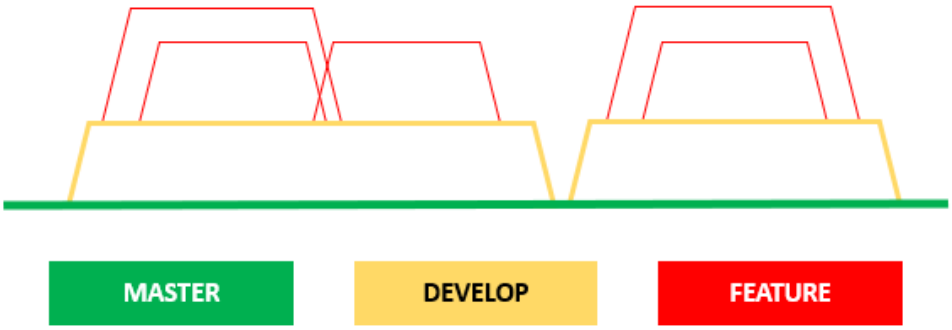

***Figure 5**. Scheme of the typical structure of Gitflow.*

4.3. SimPET file structure

After installation, which is automatically done by executing the ‘setup.py’ file, the root SimPET directory looks as show in *Figure 6*. It is important to not modify this structure, as both SimPET and STIR are already set up to read the input files and write the output files in the default directories.

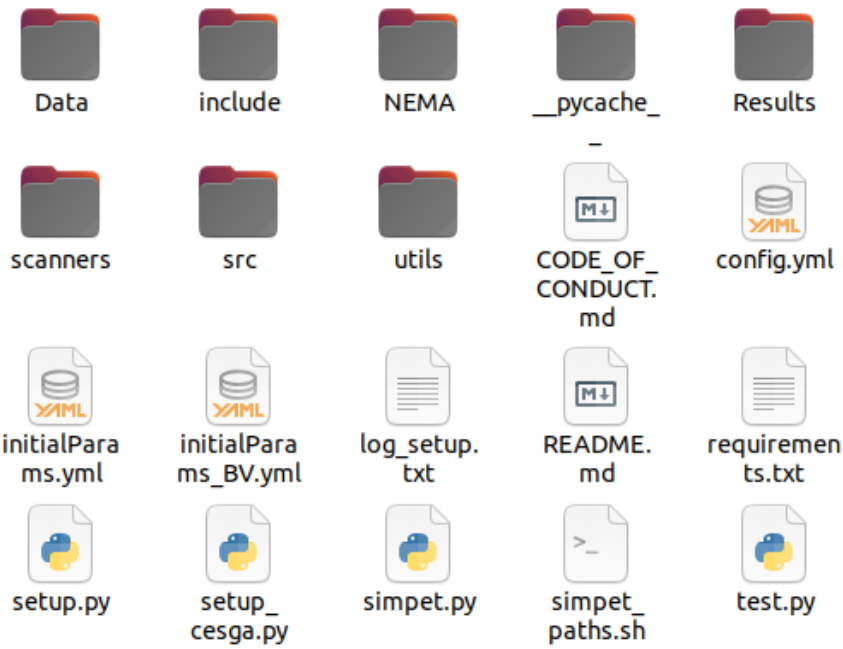

***Figure 6**. File structure of the root of the SimPET directory.*

Thus, in the Data directory there are the different attenuation and activity maps for the different simulations, in separate folders. In the parameters we will indicate these names to make SimPET know where has to read the maps. At the end of the simulation, both sinograms and reconstructed images will be stored in the Results directory, in the folder name we specified. Note that each time a simulation is executed, the results with the same name are overwritten.

In the scanners folder, the file parameters describing each of the scanner features (concerning physics, filtering, …) are stored inside the scanner folder. However, the files with the simulation parameters (time, dose, …) can be stored anywhere of the SimPET root as we have to specify its path. However, it is recommended to store them in the main root for a better organisation.

The remaining folders include program-related files with which we will not interact much during the project. Just for reference, in the *include* directory are stored the different C++ libraries needed for SimSET and STIR to work properly, in *utils* we can find some scripts implementing specific tools for SimPET, and in *src* we can find the SimPET program code itself. In the NEMA folder we will store all the elements related to this project.

The scripts (both simulations and processing of the data) have to be in .py (Python) format. As they are launched directly by console, they can be anywhere in the SimPET directory but again, it is recommended to store them in the main root. Both maps and output images are in Analyze format, which is an alternative to the NiftiFormat (.nii) relatively common in Biomedical Imaging, that consists in two files, the image file itself with the numeric matrix (.img) and the header (.hdr) containing images parameters like the size of the pixels or the typology of the image (integer, float…). Both files have to be placed together, also when visualizing them in some software. Another remarkable type of file that we are going to use are the .log files, which are text files that

record all errors, progresses and details of the simulations and reconstructions, and will help us to depurate and control the executions.

There are many other typology of files associated to the SimPET simulations (e.g., .mat files associated to the Analyse format or the .v sinograms), but they are not relevant and their handling escapes the scope of this project.

4.4. SimPET parameters

The SimPET program is designed in a way that there are a set of parameters that can be changed by modifying a list of entries. These files are in .yml format, which are simple to modify with the yaml Python library by treating them analogously as a dictionary. They have the following structure:

```
key 1: value 1

key 2: value 2

      (…)

key N: value N
```

In such a way that a certain value can be retrieved by calling the `filename[key]`. Note that this key has to be unique. Similarly to dictionaries, a new key can be opened by initializing `filename[newkey] = newvalue`.

4.4.1. Simulation Parameters

On the one hand, there are a set of parameters related to the features of the simulation. These control the acquisition (the dose administered, the time of acquisition…) and are set to be modified at every new simulation that is launched. Actually, this file is directly passed as an argument in the SimPET object that is created to generate a simulation, as follows in the following chunk:

```
import simpet
simulation = simpet.SimPET(simulation_path)
simulation.run()
```

We will mainly work by changing these parameters. The following table lists the different parameters available that can be modified. We can group them depending on if they refer to the SimPET environment, the PET system, the directories, the features of the phantoms, or the SimSET input parameters. Note that for many of the parameters their functionality may not be already available or fully deployed as a still-developing project:

Table 2. List of parameters for simulation.

Parameter	Type	Definition
Simulation environment		
sim_type	string	Program chosen to be used to simulate the data. At the date of writing, only "SimSET" is available.
do_simulation	integer	Parameter to control if a simulation has to be done or not. 1 lets the program to simulate, while 0 skips the simulation.
do_reconstruction	integer	Parameter to control if a reconstruction has to be done or not. With 1, the program reconstructs the simulated data in the Results directory, while 0 skips the reconstruction.
divisions	integer	Number of subprocesses to be launched during the simulation. That is, the number of subprocesses to be launched in parallel, dividing the simulation. This fastens the computation, but it is limited by the processor capacity of the computer.
PET system		
scanner	string	Name of the scanner that will perform the simulation. It has to coincide with one of the filenames of the scans directory, as it will read the characteristics of the scanner from there.
model_type	string	Geometrical modellization of the scanner. At the time of writing there are two types of model available: 'simple_pet' (no geometrical restrictions) or 'cylindrical'.
Directories		
patient_dirname	string	Name of the folder where the phantoms are stored. It corresponds to one of the folders in the Data directory.
act_map	string	Name of the header inside the 'patient_dirname' directory corresponding to the activity map. Note: it has to be included the '.hdr' extension.
att_map	string	Name of the header inside the 'patient_dirname' directory corresponding to the attenuation map. Note: it has to be included the '.hdr' extension.
output_dir	string	Name of the folder that will be created inside the Results directory, where both the simulation and reconstruction will be stored.
Configuration of the phantom		
center_slice	integer	Number of slice of the activity and attenuation phantoms that has to be set in the center of the FOV of the scanner. In case of 0, the central slice is set at the center (which is the usual case).
total_dose	float	Total radioactivity dose that is introduced in the patient (expressed in mCi). The dose is distributed then accordingly to the activity phantom.
simulation_time	float	Time of acquisition of the scanner (expressed in seconds). Note that this simulates the time the scanner would be acquiring in real life, but it does not express the time of simulation.
SimSET input parameters		
sampling_photons	integer	Number of decays to be simulated. It is related with the noise of the final image: the higher the value the lower the noise, but the larger the simulation.

photons	integer	Number of photons to be detected until the simulation finishes. In that case, the time of acquisition is not respected. For 0, the simulation is thus limited by the simulation_time.
add_randoms	integer	A control value to determine if random coincidences have to be considered in the simulation for more realism (1) or not (0). In case of 1, it forces both sampling_photons and photons to 0.
phglistmode	integer	A control value to determine if the files for the Photon History Generator has to be created (1) or not (0). It shall be activated only if needed, as it can occupy potentially a large space in disk.
detlistmode	integer	A control value to determine if the files for the Detector Module have to be created (1) or not (0). It shall be activated only in case of add_randoms=1.

Actually, many of the parameters will adopt a fixed value. We specify the most relevant in the table below. As a first approach, we will work with the best characterised scanner among the available ones, which is the Discovery ST from General Electrics Healthcare (there are others in process to be validated or improved, like the Advance from GE as well, the mCT/mMR from Siemens or the Vereos). Besides, we will use always the cylindrical model, as its results are expected to be more reliable in respect of real behaviour. The phantoms will be generally designed to have the transaxial FOV at its central slice. Finally, we are interested in controlling the simulation with the simulation time and not depending on how many photons were detected (thus, we are putting 0 in the photon parameters), while we will activate the random coincidences for more realism.

***Table 3**. Relation of the fixed default SimPET parameters.*

scanner	'Discovery_ST'
model_type	'cylindrical'
center_slice	0
sampling_photons	0
photons	0
add_randoms	1

4.4.2. Scanner Parameters

As we described above, one of the parameters that can be changed in the simulation is the scanner. Depending on this field, the SimPET simulation will take the parameters of the corresponding scanner. Generally all these values are maintained fixed for all the simulations of a scanner, but some corrections may need to be done, especially on the parameters affecting the filtering or correction of the reconstruction of the image. The intrinsic parameters (that is, the ones referring to the physics of the scanners) should be not changed. We can divide the parameters into different groups: Scanner Description, Crystal Description, Binning Parameters, Energy Window, Reconstruction, Attenuation Correction, and Filters, which are collected in the following table:

Table 4. List of the scanner parameters and their default values.

Parameter	Type	Description	Default value
Scanner Description			
scanner_name	string	Name of the scanner	'GE Discovery ST'
simset_material	integer	Material of the detectors, determined by SimSET's attenuation table. Usually BGO (10), LSO (18) or GSO (19)	10
average_doi	float	Depth-of-interaction measures the capacity of achieving high spatial resolution and sensitivity	0.84
scanner_radius	float	Radius of the scanner (cm)	44.31
num_rings	integer	Number of rings of the detector	24
axial_fov	float	Axial Field-of-view (that is, maximum longitudinal size the scanner is able to image).	15.7
Crystal Description			
z_crystal_size	float	Axial size of the crystal (cm)	0.63
crystal_thickness	float	Crystal thickness (cm)	3
energy_resolution	float	Measure of the fluctuations in the energy, expressed in FWHM %.	16
Binning			
num_aa_bins	integer	Number of views of the sinogram (set to be half of the detector rings).	210
num_td_bins	integer	Number of bins of the sinogram.	249
Energy window			
min_energy_window	float	Range of photon energies accepted. If the energy is outside the range, it is not detected (keV)	375
max_energy_window	float		675
coincidence_window	float	Interval of time within two photons are considered as a pair and form a LOR (ns)	11.7
Reconstruction			
max_segment	Integer	Maximum ring difference accepted	23
zoomFactor	float	Zoom applied to the X and Y directions	1.55
xyOutputSize	integer	Number of X and Y voxels of the reconstructed image	128
zOutputSize	integer	Number of slices of the reconstructed image	47
zOutputVoxelSize	float	Thickness of each slice (mm)	3.27
numberOfSubsets	integer	Number of subsets considered in the OSEM algorithm	7
numberOfIterations	integer	Number of iterations in the OSEM algorithm	32
savingInterval	integer	Determines how many intervals are saved in the Results directory	8

recons_type	string	Method of reconstruction	"OSEM3D"
Attenuation Correction			
analytical_att_correction	integer	Choose if SimSET should correct the image by attenuation (1) or not (0)	0
stir_recons_att_corr	integer	Choose if STIR should normalize the image with an attenuation map (1) or not (0)	1
analytic_scatt_corr_factor	float	Fraction of scatter photons kept	0.15
stir_scatt_corr_smoothing	integer	Inclusion of the smoothed scattered photons (1) or not (0)	0
analytic_randoms_corr_factor	float	Fraction of random photons kept	0.2
stir_randoms_corr_smoothing	float	Inclusion of the smoothed random photons (1) or not (0)	0
Filtering			
x_dir_filter_FWHM	float	FWHM of the smoothing filter	1.5
y_dir_filter_FWHM	float		1.5
z_dir_filter_FWHM	float		3
psf_value	float	PSF size of the smoothing filter	1.125
add_noise	integer	Determines if noise is added (1) or not (0) in the image	0

4.4.3. Configuration parameters

Finally, there is a third file of parameters in the SimPET directories, named 'config.yaml', which basically describe the physics of the positron annihilations and photons. For example, here it is specified the isotope used (in our case, ^{18}F) or the minimum energy accepted of the photons. As these are parameters that are associated to the particle phenomena and are not parameters that could be changed in real life, we will not work with these parameters and will remain on their default configuration.

4.5. Phantoms design

One fundamental step previous to starting with the simulations is the design and creation of the phantoms. A phantom is defined as "a real object designed to simulate the human body, or parts of it, for specific clinical conditions". In our case, the phantoms to be used are formed by two objects, activity and attenuation maps, which essentially are matrixes of voxels (with the same dimension) that are received as inputs by SimSET in order to start simulating. [25]

An attenuation map intends to reflect the different materials that form the sample of study in the scanner. Attenuation is the phenomena by which a photon loses intensity after crossing a certain material. Different materials or tissues attenuate the photons differently, so we may expect a difference of counts and consequently a contrast of intensities on the final reconstructed image. The attenuation map reflects thus how this tissues are, emulating the "physical body" that is going to be scanned. To do so, a number code is stablished in SimSET such that each voxel is associated to an integer that represents a certain material. SimSET actually accepts 30 materials (including

biological tissues, crystallographic materials, absorbers…). The materials that are going to be used mainly in this project are collected on the following table [26]:

Table 5. Relation of the most common materials and their associated index.

Material	Attenuation index
Air	0
Water	1
Lead	8
BGO	10
Perfect absorber	17
LSO	18
GSO	19
Aluminium	20
NEMA polyethilene	26

An activity map, however, does not reflect the physics of the body, but the distribution of the radioactivity concentration. With the simulation we can control the total dose injected in the scanned body, but this isotope may not be distributed homogeneously (e.g., in human body there are tissues with more affinity than others, which is the principle of the contrast). Thus, in an activity map the matrix reflects the relative concentration of activity compared to the other voxels. That is, a voxel with a 2 will have double the concentration to a voxel with a 1. With an activity map and the total dose, it is trivial to compute the dose at each voxel, something that SimSET does automatically.

The construction of a phantom may be difficult and laborious, especially with complex geometries, which is the case of the human body. Luckily, the NEMA protocol well-defines the expected phantoms and furthermore, most of them do not imply a highly-complex geometry. Following we describe each of the generated phantoms.

On the one hand, the SR test requires the definition of six point sources (i.e., one voxel) at the positions as positioned in *Figure 7*: a set of 3 at the centre of the FOV slice in (0, 1 cm), (0, 10 cm) and (10 cm, 0) positions, and another set in the same positions but in a slice displaced ¼ of the FOV.

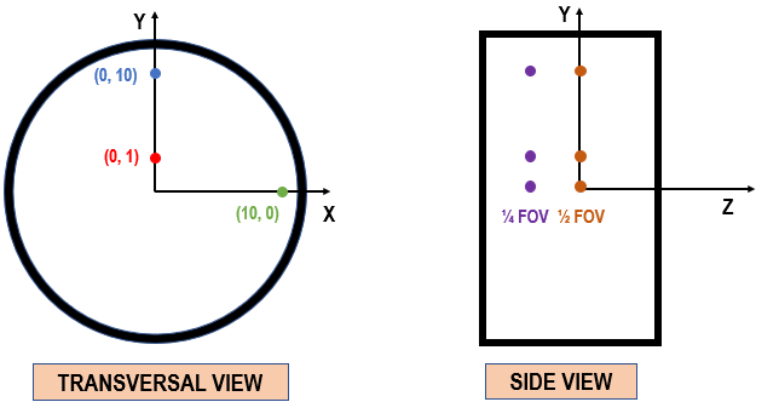

Figure 7. Transversal and side positions of the six source points for SR test.

In the case of the attenuation map, the pixel forming the point source is set with a one (representing the water where radioactivity isotopes are suspended), with all the other voxels filled with zeros representing the air. In the case of the activity map, the set value has to be non-zero, but is indifferent as all the distribution will fall to this particular voxel. In *Figure 8* we attach the activity map of one of the positions. Note that, as we are generating six different maps, we are going to launch six different simulations, which can be clearly optimised. In fact, only one map with the six sources could be generated and then work with only one image, assuming that the six sources won't affect each other. However, this might difficult the image processing step, so for simplicity we will leave the optimisation of the test as further work.

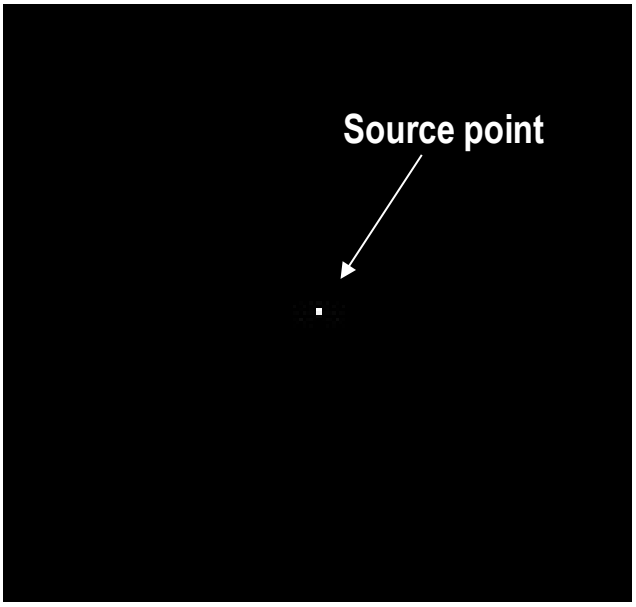

***Figure 8**. Activity map for the (0, 1) source point at the FOV.*

For the sensitivity phantom, we are asked to build a set of five aluminium phantom sleeves that are concentrically added, each one having a higher diameter than the previous one, as *Table 6* states. Note that all the sleeves have the same thickness and longitude.

***Table 6.** Diameters of the sleeves for the Sensitivity test.*

N. of sleeve	1	2	3	4	5
Inner diameter (mm)	3.9	7.0	10.2	13.4	16.6
Outer diameter (mm)	6.4	9.5	12.7	15.9	19.1

Thus, it is equivalent to build five attenuation maps with a disk filled with index 20 (representing Aluminium as an absorber) of the outer diameter size, and a second disk formed with 0 (i.e., air) of the size of the smallest sleeve inner diameter, that is, 3.9 mm. On the other hand, the activity map will be the same for all the sleeves (therefore, we are generating a unique phantom), consisting on a line source of one voxel diameter with the same axial longitude of the sleeves. Thus, as we will expect an homogeneous activity distributed along the phantom, the value of activity can be arbitrary but the same in all the line. An example of one of the attenuation and activity map generated is shown in *Figure 9.*

Figure 9. *Sleeve 5 attenuation and activity maps.*

Next, the IQ phantoms are without doubt the more complex of all the considered. The idea is to emulate the torso of a patient, with very precise specifications. The phantom has to be at least 180 mm long. Six spheres of increasing diameters (10, 13, 17, 22, 28 and 37 mm) have to be centred at the same plane. Their centres have to be equispatially positioned on a circumference of radius 114.4 mm. At the centre of the phantom, a tube has to be positioned longitudinally emulating the attenuation of the lung.

In this case, the phantoms were already provided by the development team, so there was no need to generate them. The three views of both maps are presented in *Figure 10* and *Figure 11*. Thus, the attenuation map basically has to resemble the complex geometry of the torso, by filling with 26 (corresponding to polyethilene, as stipulated by the NEMA) the voxels corresponding to the walls. The interior part of the phantom and the spheres is besides filled with water. In the case of the activity map, it takes the value of the background and multiplies it by a certain factor (usually 4 or 8) to determine the activity inside the four smaller spheres (hot spheres). The activity of the two biggest spheres is set to 0 (cold spheres).

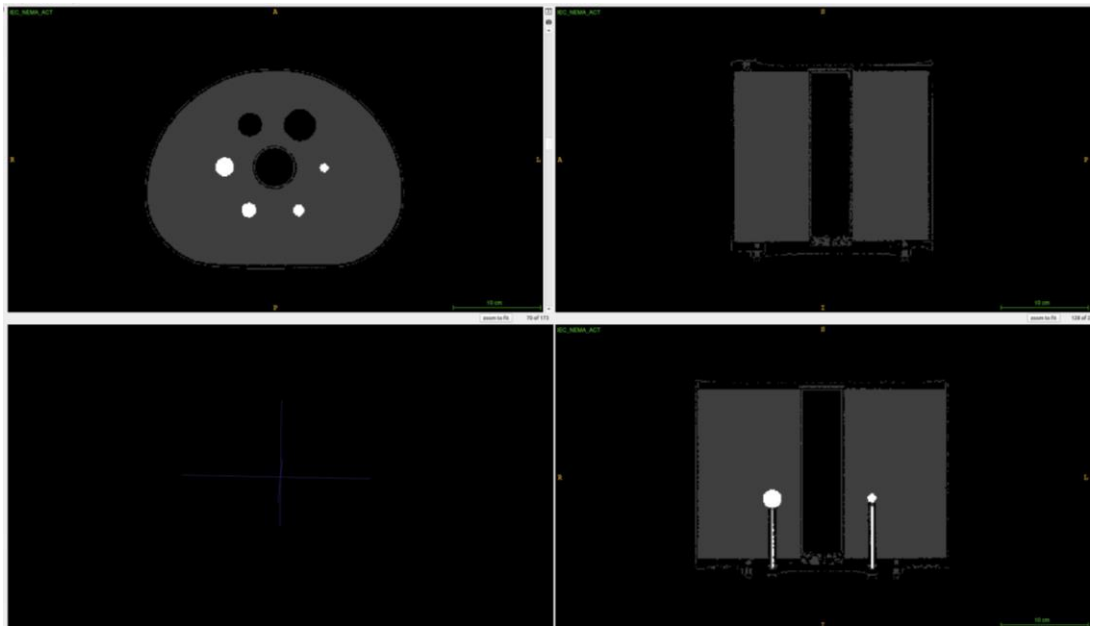

Figure 10. *Three views of the IQ activity map.*

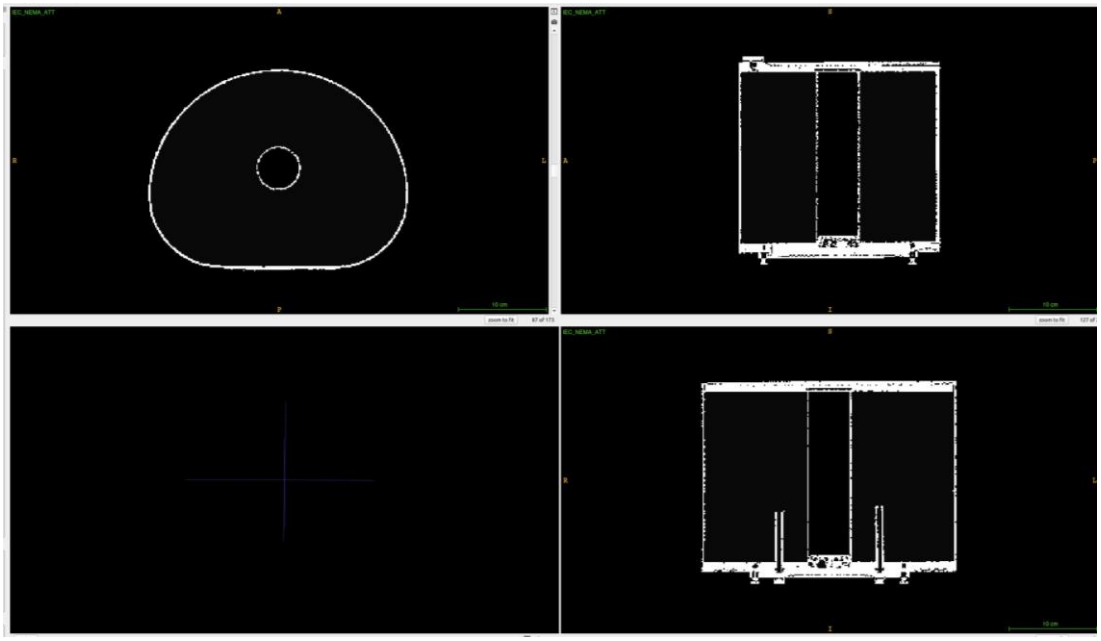

Figure 11*. Three views of the IQ attenuation map.*

Finally, for the Scatter Fraction and the Accuracy tests, the same phantom can be used. In this case, it consists on a polyethylene cylinder, with a hole axially traversing the cylinder positioned 45 mm of the FOV. Thus, the attenuation phantom simply consists on a cylinder made of the NEMA polyethylene, except for the discentered hole, filled with water. The activity phantom is then similar to the Sensitivity one, with a line source positioned in the hole of the activity map. The section of both maps is shown in *Figure 12.*

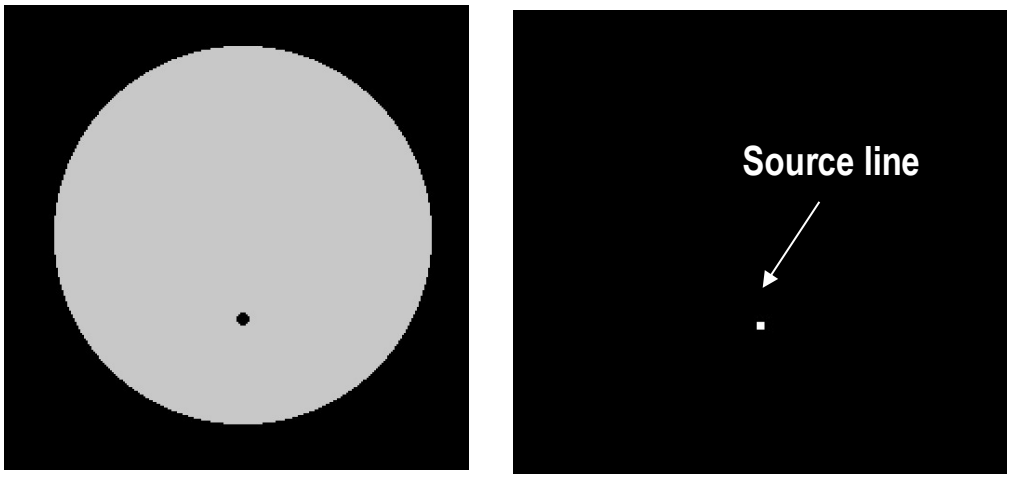

Figure 12. *Scatter fraction attenuation (left) and activity (right) maps.*

4.6. IQ Spheres Segmentations

In the case of the Image Quality test, a previous segmentation of the obtained image is required before start computing the results. In this case, we need to draw ROIs in the slice imaging the centre of the spheres, one ROI per sphere. Thus, as can be seen *Figure 13,* the problem to solve is to find the six centres of spheres of different diameters, that form a regular hexagon. First of all, it was convenient to mask the image into three regions: background of the image, background of the phantom, and spheres, which can be easily thresholded as their characteristic intensities are clearly different: the background has very low intensities, while the spheres present very high values. In particular, the lung insert at the centre has similar intensities as the background of the image, so we can positionally restrict the more central pixels (e.g., the second third of the image) and mask them separately from the background. The obtained masking of the image can be seen in *Figure 13*, where by visual inspection can be appreciated how the spheres get dilated in the mask in respect to the original image, so the masking might be refined in the future.

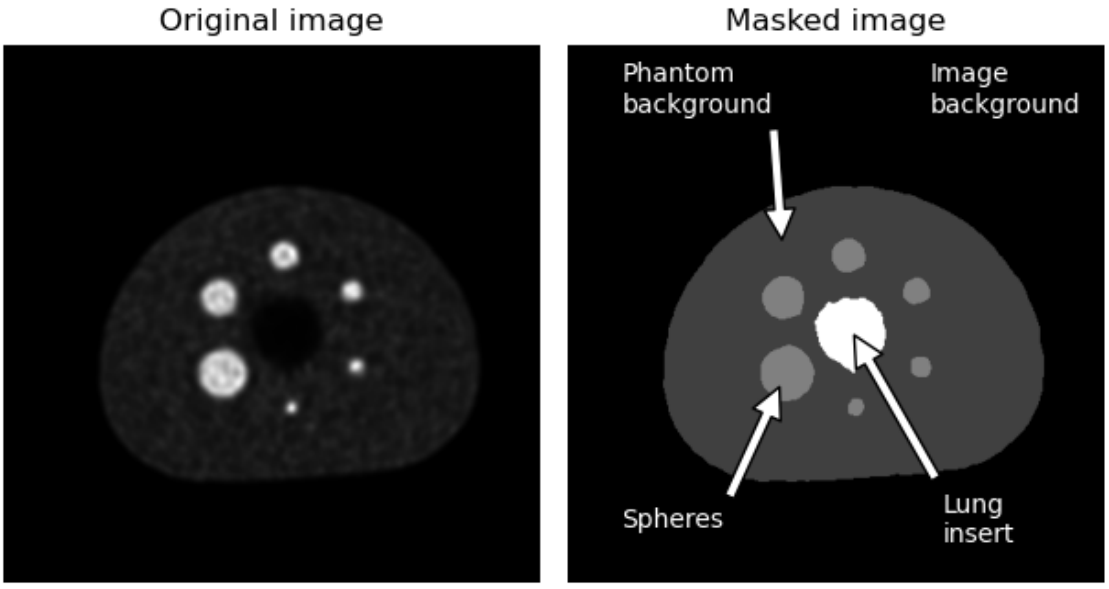

Figure 13. *Original PET image and their corresponding masked image.*

The straightforward approach, but also the most simplistic, is to manually segment the spheres with a software like 3DSlicer. These programs offer tools that would allow to delimit the regions with relatively speed, and without requiring much prior knowledge from the user. However, this approach lacks of generalization and automatization, meaning that before executing the test it would be required to perform this step, which goes against one of the main objectives of this project, which is precisely the automatization. Thus, this approach is discarded. Another option consisting in manually setting the centres of the spheres and drawing circles from there with the tools that the library *skimage* offers is also not considered for the same reasons.

Next, another possibility dealed with the computation of centroids, which given a distribution of indexes can be computed as the mean of their coordinates. Thus, it firstly took the centre of the torso as the centroid of the lung insert, from which, taking advantage of the regular hexagon the centres of the spheres are supposed to form, traced six lines in the masked image dividing the space in six regions, with one sphere per region, as can be seen in *Figure 14,* where can be seen that each sphere can then be segmented individually. So, as we now had six sets of indexes, we computed individually each centroid and assigned them as each sphere centre. This method is robust as each spheres has statistics enough to compute the centroid, but it is also sensible to possible geometrical irregularities of the phantom.

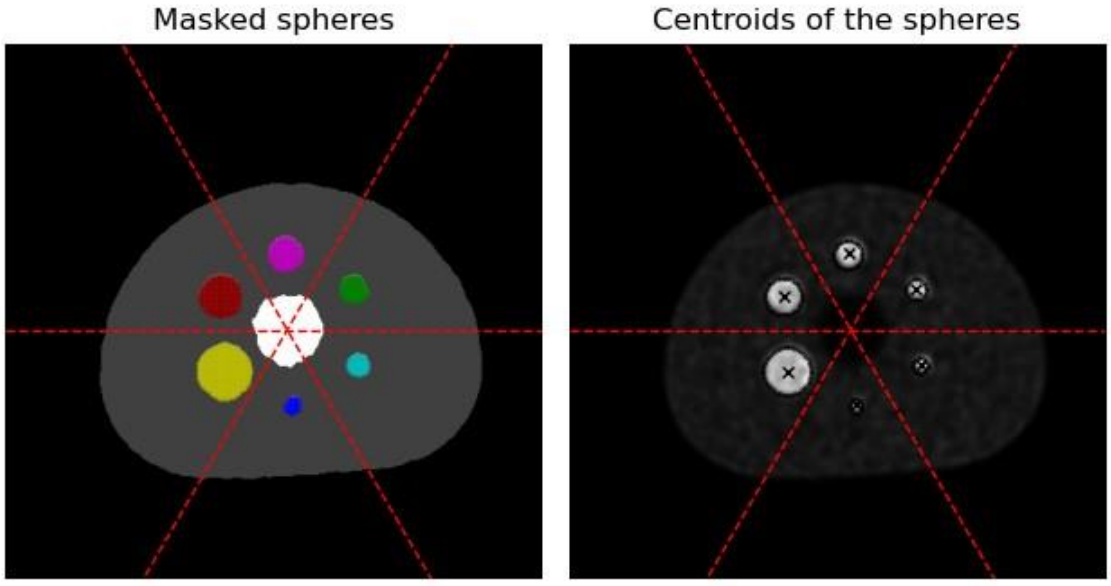

Figure 14. *Segmentation of the spheres with the centroids method (left) and the resulting ROIs (right).*

The next alternative worked not with the spheres individually, but as a collective. In this case, knowing the diameter that the centres of the spheres form (114.4 mm as stipulated by the NEMA), we built a circumference with six equispaced vertexes (forming a regular hexagon). Thus, we needed to find the best position for the centre of the circumference. To do that, we iterate for each lung insert index, and we count the number of voxels corresponding to the spheres, choosing the centre as the position with the highest count, as we are assuming here that the centre is the position more coincident with the spheres. This method has the advantage to work with the six spheres collectively, which may preserve the distances from the phantom, but it may result uncentered if

the mask shapes are not much regular. In *Figure 15 (left)* can be observed the centres positioned in the image and the circumference they form.

Another alternative could be based by the assumption that the centre of the spheres is located in the maximum intensity point of each sphere. However, theoretically this is true only in the source point case, a scenario that may not be approximated in the spheres, especially for the bigger ones. In any case, in *Figure 15 (right)* can be seen the result of centring the spheres in the maximum intensity points. As can be seen, the chosen points are arbitrarily situated, especially in the biggest spheres, are arbitrarily uncentered. One improvement could be performing a weighted sum on a NxN kernel size. However, we tried for N=3, N=5 and N=7, and the changes were minimum, of one or two displaced voxels.

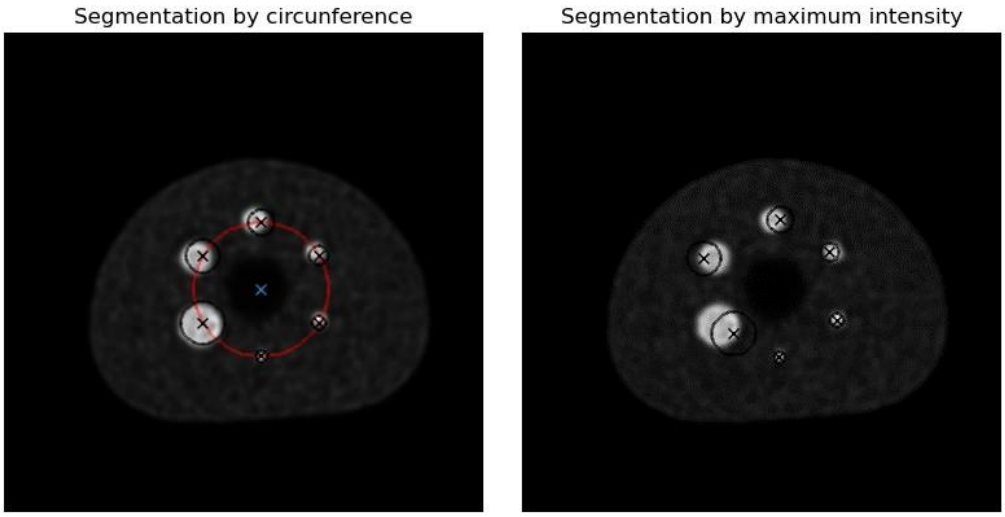

Figure 15. *Segmented ROIs by the circumference (left) and maximum intensities (right) methods.*

Entering in more advanced methods, it was suggested to use clustering methods to group the spheres, instead of the tracing of lines. This method would be suitable for more complex geometries, or for phantoms in which we don't have many information. However, as we have enough information and the distribution of spheres is regular enough, we discarded the use of clusters as it would require some time of learning. Besides, our suggested method aboved has shown to be working enough.

Finally, resampling techniques could also been used, but again would have required some time of knowledge. The idea would have been to resample the positions in the phantoms, which can be easily found as are created by ourselves, to the reconstructed image.

Thus, from the methods we did not discard (circumference, centroids and maximum intensities), by visual inspection it seems to work better the centroids positioning, especially compared to the maximum intensities. Nevertheless, it is convenient to use some figure of merit to objectively select a method. In our case, we defined two figures of merit: the number of voxels corresponding to the spheres in the mask, and the mean intensity of the defined disks, which we want to maximise in both cases:

Table 7. Merit figures for the three proposed segmentation methods.

	Max. Intensities	Circumference	Centroids
Sphere voxels	1755	2236	2513
Mean intensity	68708	75325	83419

The *Table 7* is a confirmation that the method that may suit better this test is, consequently, the use of centroids, which is indeed the chosen method we will implement.

5. Detailed Engineering

In this section, the implementation of each of the NEMA tests is specified, by determining the parameters finally selected and the flow diagram of the final code. In all the cases, the used scan to test the algorithms will be the Discovery ST. All the derived expressions in this section are extracted from [4].

5.1. Spatial Resolution

In this first test, we are required to compute the full width at half maximum (FWHM) and the full width at tenth maximum (FWTM) of the point source response, that is, to compute the width (in cm) of the distribution of intensities in the reconstruction of the image of the crossing points at the half and one tenth of the maximum of the peak. A conceptual example of the computation of these values can be seen at *Figure 16*. From these values, as it will be discussed later on, we can extract the transversal and tangential resolutions.

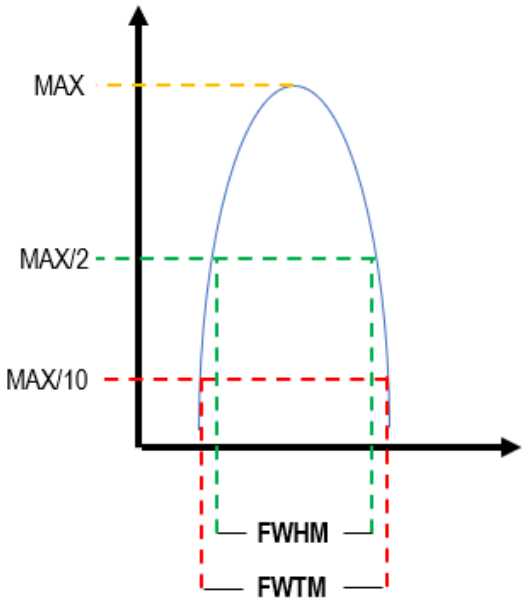

Figure 16. Scheme of the definition of FWHM and FWTM.

Firstly, a directory in the Data root is created in order to store the phantoms in the case it didn't exist yet. After that, the parameters file is modified to the following values:

Table 8. Chosen parameters for the SR test.

Parameter	SR Value
Divisions	6
Reconstruction type	FBP2D
Simulation time (s)	60
Total dose (mCi)	0.02

Next, one of the phantoms is selected (initially, the x=0cm, y=1cm at the FOV of the scanner). The directory of the phantoms is changed, and the simulation is executed. With the obtained image (in this case, a 2D Filtered Back-Projection) we extract the position of the voxel with the maximum intensity, which will be assumed to be the original position of the point source. Note that, due to the binning on the reconstruction (which implies a modification on the sizes of the voxels) the original position of the source may be displaced one voxel, but for simplicity we are assuming in all the cases we will consider only the maximum intensity voxel.

Following, three profiles are traced by retrieving the intensity along the x-, y- and z-directions, as shown in *Figure 17*, so in each of the profiles the peak should be positioned on the number of pixel corresponding to the maximum found.

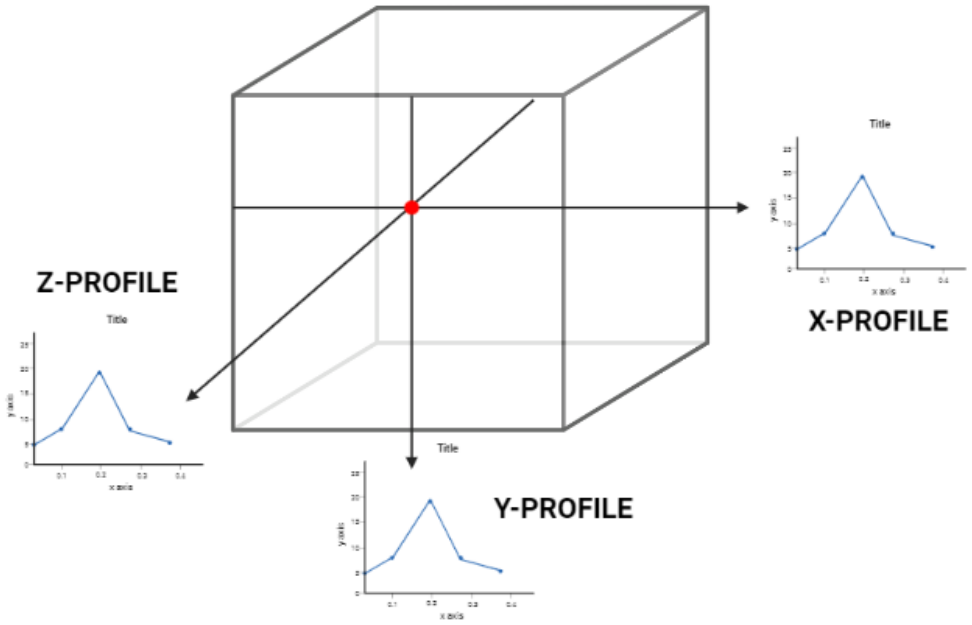

***Figure 17**. Scheme of the intensity profiles from the source point.*

The next step consists on determining the maximum of the peak. As the NEMA protocol clearly states, this maximum should not be directly taken from the profile curve, but it must be extracted from as the peak of a parabolic adjust between the maximum and the two adjacent points of the profiles. An schematic representation of this adjust can be found at *Figure 18 (left)* for more clarity.

From this maximum, the half and tenth values have to be interpolated across the profile using linear interpolation, that is, using the immediate anterior and posterior points to this value, defining them as (x_0, y_0) and (x_1, y_1) respectively:

$$\frac{y - y_0}{x - x_0} = \frac{y_1 - y_0}{x_1 - x_0}$$

It can be interpolated the inner point in the interval $(x, y) \mid x \in [x_0, x_1] \; and \; y \in [y_0, y_1]$, being y the half- or the tenth-maximum depending if we are computing the FWHM or the FWTM, respectively. Again, for more clarity a schematic representation of the process can be found at *Figure 18 (right)*.

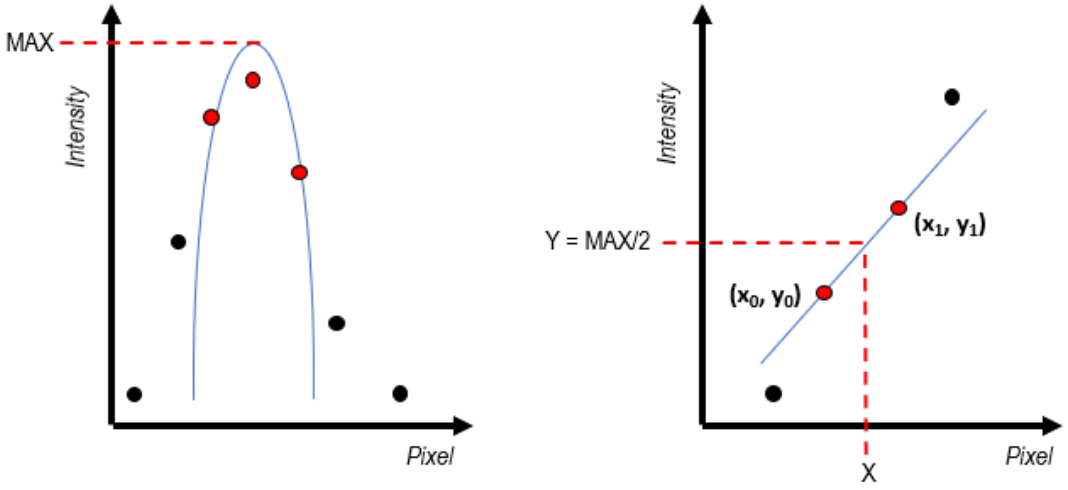

***Figure 18**. Scheme of the parabolic adjust (left) and the interpolation process (right).*

After both points (at both sides of the maximum of the parabola) are interpolated, the resolution along the axis i can be simply computed as,

$$R_i = \left(x_{right} - x_{left}\right) \cdot L_{pixel}$$

where L_{pixel} is the size of the pixel at the i-direction.

After we iterated for all the 6 source points we defined at *Concept Engineering* note that we will have 36 resolution values, 3 for each direction of the 6 source points and for both the computation of FWHM and FWTM. Thus, we need these results, that we will note with $R_i(x, y, z)$ for the i-direction to compute the resolution values. The transversal resolution is computed as the mean resolution at the x and y directions of the image. In the case of radius 10 cm, as we have a double measure, this transversal resolution can be computed in the tangential or the radial direction (that is, in the radius or the perpendicular directions, respectively). The axial resolution is the resolution at the z direction of the image. In both cases, we average the measures at the central and quarter of the FOV. These measurements are summarized in the following table:

Table 9. Set of formulas for the different resolutions.

Resolution measure	Radius (cm)	Formula
Transverse	1	$\frac{R_x(0,1,FOV) + R_y(0,1,FOV) + R_x\left(0,1,\frac{FOV}{4}\right) + R_y\left(0,1,\frac{FOV}{4}\right)}{4}$
Axial	1	$\frac{R_z(0,1,FOV) + R_z\left(0,1,\frac{FOV}{4}\right)}{2}$
Transverse radial	10	$\frac{R_x(10,0,FOV) + R_y(0,10,FOV) + R_x\left(10,0,\frac{FOV}{4}\right) + R_y\left(0,10,\frac{FOV}{4}\right)}{4}$
Transverse tangential	10	$\frac{R_x(0,10,FOV) + R_y(10,0,FOV) + R_x\left(0,10,\frac{FOV}{4}\right) + R_y\left(10,0,\frac{FOV}{4}\right)}{4}$
Axial	10	$\frac{R_z(10,0,FOV) + R_z(0,10,FOV) + R_z\left(10,0,\frac{FOV}{4}\right) + R_z\left(0,10,\frac{FOV}{4}\right)}{4}$

The values in the previous table are the ones required to be reported by the NEMA protocol, for both the FWHM and the FWTM, retrieving a total of 10 resolutions. Finally, the diagram flow of the implementation is shown:

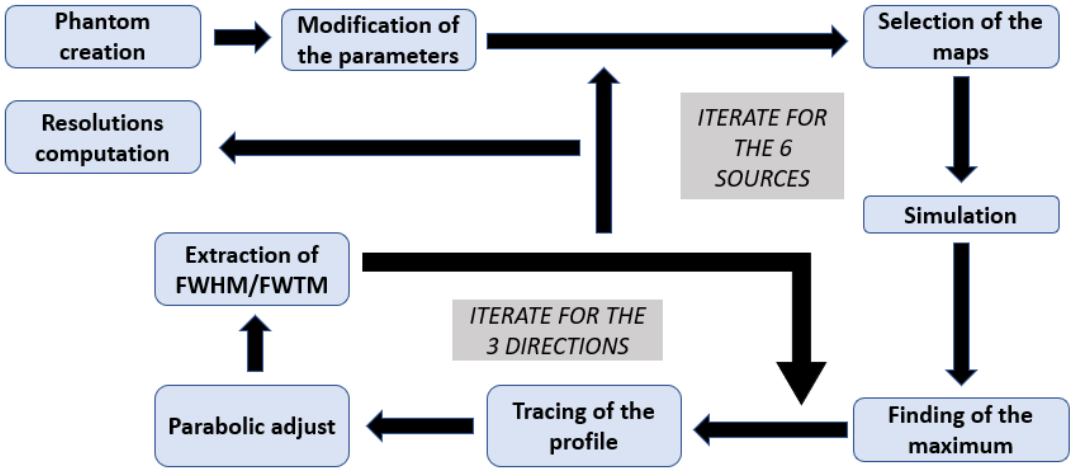

Figure 19. Diagram flow of the SR implementation.

5.2. Sensitivity

In this second test, we are asked on the one hand to compute the total system sensitivity (at both the centre of the FOV and at a radial offset) and on the other hand, to plot the axial sensitivity profile of the system, that is, the "fractional" sensitivity respect to the total sensitivity value.

5.2.1. Sinogram direct reconstruction

Before the calculation of the sensitivity, some previous appreciations are nevertheless needed. The sensitivity is a magnitude expressed in counts·[t]$^{-1}$·[Rad]$^{-1}$, where [t] represents a unit of time and [Rad], a radioactivity unit (usually expressed in Becquerels). Hence, on the reconstructed image we are not interested in the intensity (which is related to the count values, but normalized in some way) but on the number of counts detected by the scanner. This information is contained in the sinograms, which SimPET actually builds after the simulation. Actually, one of the more practical features of SimPET is that it directly splits the sinogram counts into the three types of coincidences: true, scattered and random, with a total sinogram as the sum of the three as well.

If we inspect the object corresponding to the sinogram, we observe it adopts the shape of a tridimensional matrix of elements m_{ijk}. [27] For each slice k of this matrix, the i represents the radial distance of the annihilation from the centre of the scanner, while the j indexes the different projections (i.e., the angle turned through the scanner). Thus, we have slices for each of the pair combination of rings, ordered from the highest difference from Ring 2 to 1, to the highest difference from Ring 1 to 2, that is for N rings:

$$(1, N); (1, N-1); (2, N); (1, N-2); \dots; (N-1, 3); (N, 2); (N, 1)$$

We are asked by the NEMA guidelines to perform a FBP reconstruction on our simulated sinograms. Thus, we need to manually apply the reconstruction algorithm to build each of the slices, which are set to be $2N - 1$. In the tridimensional case (FBP3D), each slice corresponds to the sum of a certain set of sinograms. If we define the Michelogram as a matrix of sinograms (whose rows and columns represent the number of the first and second ring), the sinograms are thus summed in diagonals (for more clarity, see *Figure 20*). Note that the extreme slices will contain only one sinogram (the pairs $(1, N)$ and $(N, 1)$, respectively), the second and penultimate slices will correspond to the sum of two pairs each and successively, until the central slice is reached, with a sum of N sinograms. This fact explains why the profile of the counts per slice adopts a triangular form, as we keep summing more sinograms until the central slice, where we start to concatenate less sinograms, and the lower number of counts correspond to the extremes (with only one sinogram), as we justified before. In any case, the number of counts can be computed as simply summing the values of the summed sinogram.

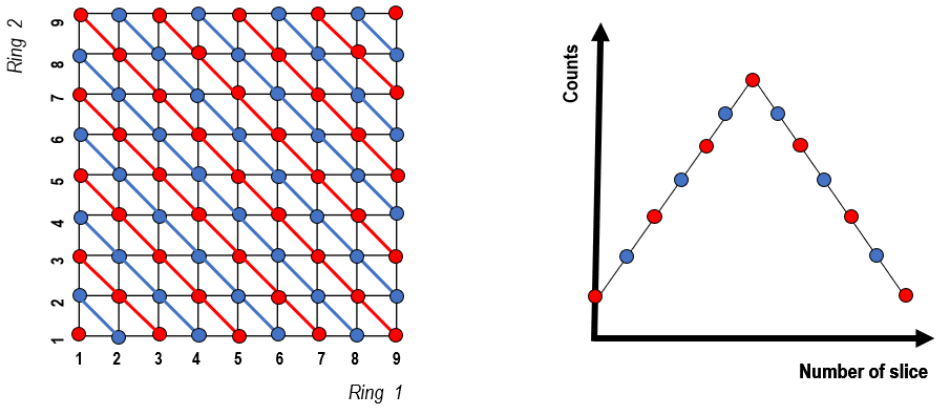

Figure 20. *Scheme of the summation of sinograms in the 3D FBP.*

In the bidimensional case (FBP2D) we are adding some restrictions. Purely speaking, in 2D it would only be considered the sinograms corresponding to the pairs formed by the same ring (i.e., $(1,1), (2,2), \dots, (N,N)$). Nevertheless, it is common to relax these restrictions and admit the LORs detected in rings of a certain difference. This feature is controlled by a parameter called span (let us note it as S), which is expected to be the sum of the number of sinograms considered in an odd and an even slice. It can be shown that thus the maximum difference Δ between rings admitted is $\Delta = \frac{S-1}{2}$. Thus, in the central slices we keep summing $\frac{S-1}{2}$ and $\frac{S+1}{2}$, which forms in the count profile a characteristic ridged shape. In *Figure 21* can be seen an example of the reconstruction for S=5 and S=7, where it can be appreciated that this reconstruction method is equivalent to the tridimensional one by "cutting" the edges of the Michelogram. In fact, FBP3D can be seen a particular case of FBP2D with $S = 2N - 1$, because as we keep increasing the span number, the number of ridges and their heigh is decreased until reaching this triangular shape in the limit.

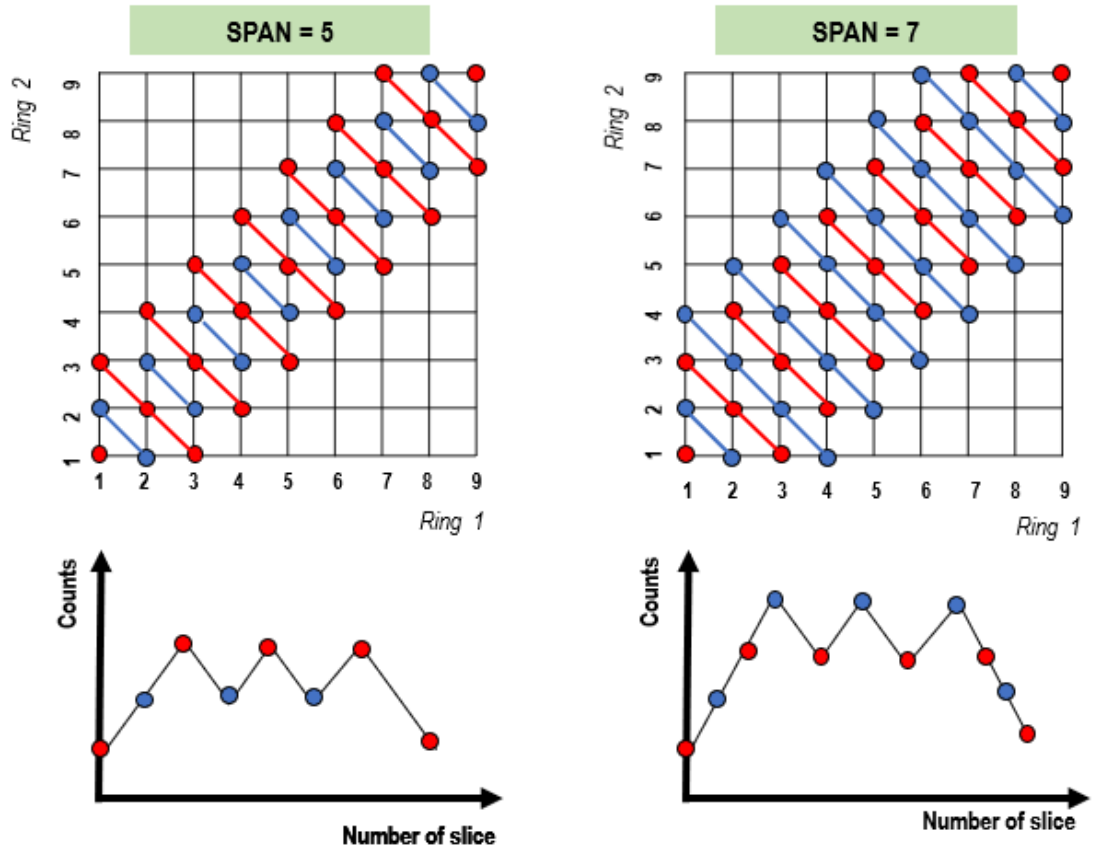

Figure 21. *Scheme of the summation of sinograms in the 2D FBP (for spans 5 and 7).*

Consequently, we proceeded to implement two Python functions that were able to receive a stacked matrix of sinograms, like the ones computed with SimPET, and extracts a new matrix with the corresponding sinogram sums. Note that we are converting a matrix from N^2 to $2N + 1$ slices. Following, the pseudocode of the function is introduced:

```
sinogramator(scanner, path of the sinograms, span)

"""Arguments:

      scanner: Name of the scanner used ('Discovery_ST',…)

      path of the sinograms: directory where the sinogram of interest is
      stored (usually, in Results directory).

      span: number of span for FBP2D. If 0, computes FBP3D.

      dose: dose stablished in the simulation (mCi).

      Time: time of the acquisition (s)."""

1. Determine the path of the scanner file and read the number of rings
(N).

2. If span is 0, span = 2*N-1 (3D case).

3. Build a list indexing the sinograms depending on their ring difference
[1, 2, …, N, …, 2, 1].

4. Read the sinogram matrix.

5. Build a list of tuples with the pairs of rings considered for each
slice. [[(1, 1)], [(1, 2), (2, 1)], … ].

6. For each number of slice (Ns = 2*N-1):

      For each tuple in element tuplelist[Ns]:

            7. Compute the ring difference as the difference of the
            elements of the tuple.

            8. Define a shift number that consists on the index where
            the sinograms of the computed ring difference are (sum of
            the indexing positions) plus the number of already used
            sinograms of this ring difference (using a counter).

            9. Update the counter

            10. Sum the sinogram values

      11. Store the total number of counts.

12. Normalise the number of counts per time (s) and activity (kBq).

13. Return the normalised values.
```

Note from the previous function, that we integrated both reconstruction methods into one as we justified before the relation between FBP2D and FBP3D.

5.2.2. Total Sensitivity and Profiling

With a function that returns the profile of counts of the sinogram, it is now simpler to compute the sensitivity of the system. Similarly to the spatial resolution test, a directory in the Data root is created in order to store the 5 sleeve phantoms in the case it didn't exist yet. After that, the parameters file is modified to the following values:

Table 10. *Chosen parameters for the sensitivity test.*

Parameter	SR Value
Divisions	6
Reconstruction type	FBP2D
Simulation time (s)	30
Total dose (mCi)	0.10

Next, a for loop iterates for each of the sleeves. For each one, extracts the sinogram of the total counts and, by using the 'sinogramator' function, extracts the number of counts for both FBP2D and FBP3D. For each slice j and sleeve i, we are asked to compute the count rate, which is defined as,

$$R_{ji_{CORR}} = R_{ji} \cdot 2^{\frac{T_j - T_{cal}}{T_{\frac{1}{2}}}}$$

where T_{cal} is the time recorded at the start of the acquisition, T_j the time of acquisition and $T_{\frac{1}{2}}$ the period of semidesintegration (that is, $T_{\frac{1}{2}} = 110\ min$). As we are simulating regular acquisitions of $T_{acq} = 30\ s$, note that this factor is actually negligible, with a value of about 1,003: consequently, in this case we are assuming that $R_{ji_{CORR}} \approx R_{ji}$. From this value, we can simply extract the count rate of the sleeve with $R_j = \sum_i R_{ji}$, that is, by summing for all the count rate slices.

Besides, we are required to represent and exponentially fit our data under the following regression equation:

$$R_j = R_0 \cdot \exp\left(-\mu_M \cdot 2 \cdot X_j\right)$$

where R_0 is the count rate for a thickness of 0, as the value we want to extrapolate, and $-\mu_M$ os the corrected metal attenuation coefficient, which acts only as an auxiliar parameter and we are not interested in its value. On the other hand, X_j is the accumulated thickness of the sleeves, and as stablished for the NEMA protocol, each sleeve has the same thickness of 2.5 mm, meaning that the accumulated thickness is actually a linear relation: $X_j = 2.5 \cdot j\ (mm)$. We opted for directly representing the logarithmic count rates, that is,

$$\log\left(R_j\right) = -2\mu_M X_j + \log(R_0)$$

So for a linear adjust $y = mx + n$ we can directly extrapolate the 0-thickness value relating it with the independent term of the fit, $R_0 = e^n$. Finally, the total sensitivity can be retrieved by normalising this value to the initial activity of the line source (A_{cal}):

$$S_{tot} = \frac{R_0}{A_{cal}}$$

Using this value we can also fulfill the other requirement of the Sensitivity test, consisting on plotting the profile of the sensitivity for each slice, in the particular case of the thinnest sleeve. This profile can be obtained by normalising the count rate of each slice by the total count rate:

$$S_i = \frac{R_{1,i}}{R_1} \cdot S_{tot}$$

Note that the value is also multiplied by the total sensitivity, expressing then this quantity as a fraction of S_{tot}. Both results (S_{tot} and S_i) shall be obtained for the bidimensional and tridimensional cases. The following diagram flow summarises the implementation of the test:

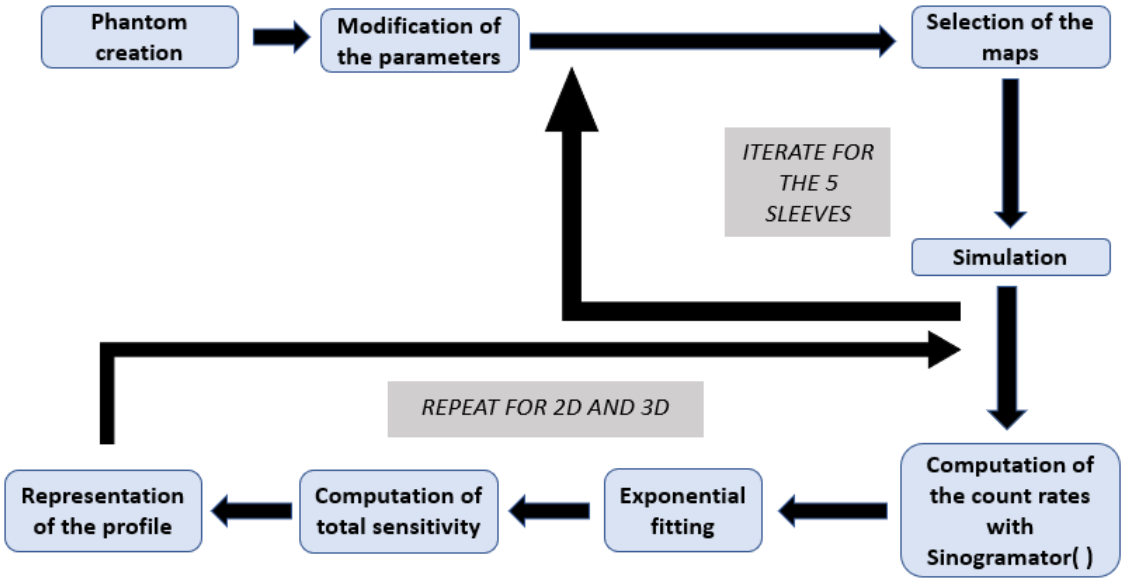

Figure 22. *Diagram flow of the Sensitivity implementation.*

5.3. Image Quality

As its own name describes, this test asks to extract some parameters from the reconstructed image to study if its quality is acceptable for clinical purposes and valid for a diagnostic. The main idea is to use the torso phantoms described in the previous section with a spheres/background concentration ratio of L/B = 4 and L/B = 8. The main limitation here is the need of free-space needed to store all the files during the reconstruction. Both doses, time and map complexity are so high that with the available space in the working computer, SimPET was not able to complet even the 20% of the simulation. Thus, in order to be able to try our program for this test, we used an example image provided by the Nuclear Medicine Service, as yearly the PET scanners need to pass a maintenance which uses a similar phantom. However, in this case the image has not 4, but 6 of the spheres filled with radioactivity, while there is no cold spheres. Consequently, in the future our program will need to be slightly modified to adapt these changes. Besides, we have at our disposal only an image with L/B=4, but this only changes in that we will execute not twice, but only once our program.

So, as we commented we are going to use a sample image, but in case of simulating these would be the recommended parameters for the simulation, similar to a real patient acquisition, noting that both time and dose are significantly higher than in the sensitivity and SR tests.

Table 11. *Chosen parameters for the IQ test.*

Parameter	SR Value
Divisions	8
Reconstruction type	OSEM3D
Simulation time (s)	300
Total dose (mCi)	10

First of all, we are asked to compute the recovery coefficients Q_H (hot spheres) and Q_C (cold spheres). To do so, we need the mean intensity of the voxels contained in the different spheres, so we proceeded to segment them with the method we chose in the Concept Engineering. We started by masking the image depending on the intensity values. The conversion is collected in *Table 12*:

Table 12. *Thresholding for the masking of the IQ image.*

Region	Image intensity value	Mask value
Image background	< 1000 (outside region)	0
Phantom background	[1000, 4000]	1
Spheres	> 4000	2
Lung insert	< 1000 (central region)	3

Note that the low intensity regions are splitted between the central and outside zones. This is to discern between the background of the image and the lung insert. Of course, these thresholds may be tuned and more crucially, parametrized to generalise it for any image (for example, using the standard deviation of the intensity distributions). A correct election of these values is really important, as otherwise the sphere zones may be distorted (by shrinking or dilating).

Thus, we proceeded to iterate over all the slices of the voxel matrix and find the one containing the sphere centres, which are assumed to be in the same plane. The condition we imposed was to keep the slice with the more number of voxels corresponding to the spheres in the mask. In the selected slice, we took only the lung insert pixels and computed its centroid as the mean of all coordinates, which from now on is assumed to be the centre of the phantom. Next, we traced the horizontal line passing through the centre, and the two crossing lines rotated 60º and 120º, in order to divide the space in 6 regions, one per sphere. Taking the notation from *Figure 23* we segmented the images collecting the corresponding masked pixels (i, j) under the following restrictions:

$$s_1: j < r_1 \,\&\, j < r_2; \quad s_2: j < r_0 \,\&\, j > r_2; \quad s_3: j > r_0 \,\&\, j < r_1;$$

$$s_4: j > r_1 \,\&\, j > r_2; \quad s_5: j > r_0 \,\&\, j < r_2; \quad s_6: j < r_0 \,\&\, j > r_1;$$

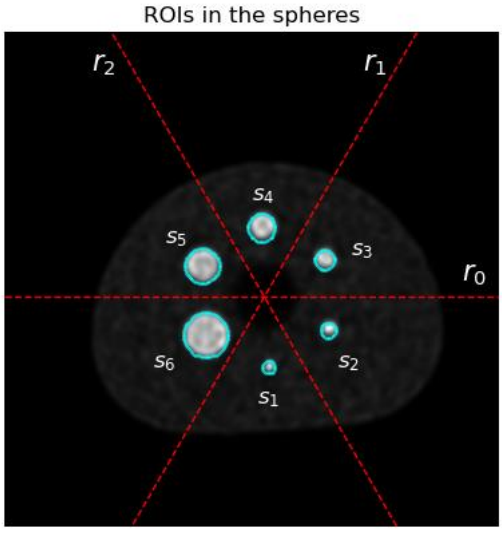

Figure 23. *ROI segmentation of the spheres and notation of the elements.*

After taking the mean value of the spheres, $C_{H,j}$ $(j = 1, \dots, 6)$, we are required to draw twelve spherical ROIs of the size of the biggest sphere (that is, 37mm diameter) in the phantom background. These spheres shall not overlap each other and should not have contact with the radioactive spheres. This procedure has to be repeated for slices at a distance of ±10cm and ±20cm, resulting in a total of 60 spheres. An example of this task is shown in *Figure 24 (left).*The respective sets of 60 spheres shall be repeated for each of the sphere sizes, positioning them in a concentric way in respect to the 37 mm ROIs. Thus, the mean of each of the 60 sets, $C_{B,j}$ $(j = 1, \dots, 6)$, must be computed.

One possibility to create the ROIs could have been drawing them manually, but apart from being an excessively cumbersome task, it lacked from automatization, which is the objective of the project. Thus, we made use of the Python random generator number to start prompting random image positions. Hence, we imposed the condition that this position should be in its totality in the phantom background masked zone, and any of the index could have been selected before (in order to avoid the overlapping of ROIs).

With these computations, we can already retrieve the recovery coefficients, $Q_{H,j}$ for each sphere,

$$Q_{H,j} = \frac{\frac{C_{H,j}}{C_{B,j}} - 1}{\frac{a_H}{a_B} - 1} \cdot 100\%$$

where $\frac{a_H}{a_B}$ is the ratio of activity concentrations between the spheres and the background and is determined during the preparation of the phantom. Analogously, the recovery coefficient of the cold spheres, $Q_{C,j}$ could be computed with:

$$Q_{C,j} = \left(1 - \frac{C_{C,j}}{C_{B,j}}\right) \cdot 100\%$$

With the obtained data we are also in position of computing the background variability of each sphere, N_j $(j = 1, \dots, 6)$ by iterating for each background ROI,

$$N_j = \frac{SD_j}{C_{B,j}} \cdot 100\% = \frac{\sqrt{\frac{\sum_{k=1}^{K}\left(C_{B,j,k} - C_{B,j}\right)^2}{K-1}}}{C_{B,j}}$$

in which, as our set has 60 ROIs, we are taking K = 60. Finally, we need to draw another set of ROIs, one in each of the five slices considered, of 30 mm diameter, centred in the lung insert, whose positions were already obtained. Again, we were asked to extract the mean intensities contained inside the ROI, C_{lung}, to obtain the relative error between the background and the insert for each slice and its mean, ΔC_{lung}:

$$\Delta C_{lung,i} = \frac{C_{lung,i}}{C_{B,i}} \cdot 100\%$$

The drawn ROI for in the lung insert in the central slice is shown in *Figure 24 (right)*.

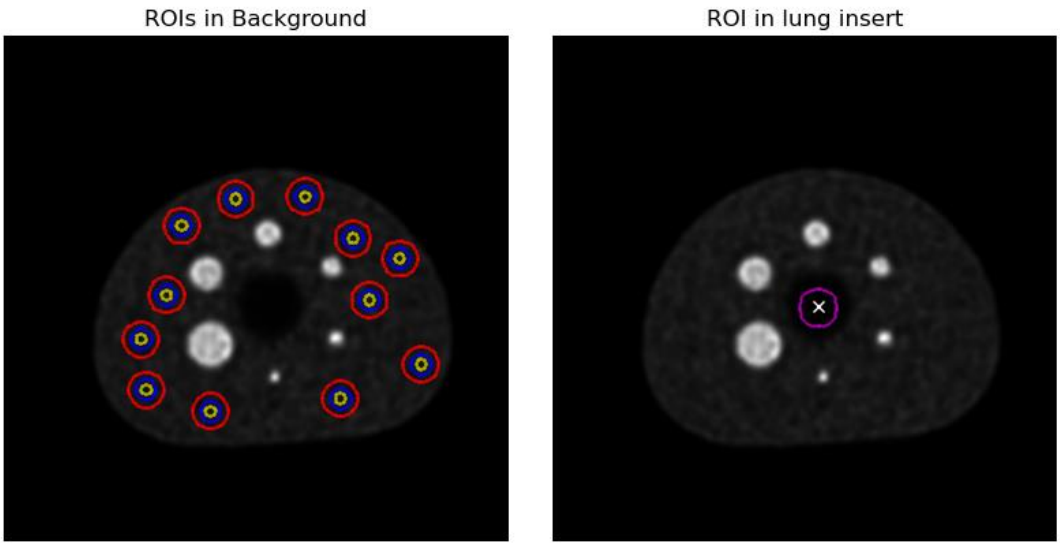

***Figure 24**. ROIs in the background (left) and the lung insert (right) of the central slice.*

5.4. Scatter Fraction

Finally, in the Scatter Fraction test we are asked to perform some measures concerning the proportion of scattered photons in respect of the total ones that an acquisition has. To do so, we are asked to take several measures for different mean activity values, defined as:

$$A_{ave} = \frac{A_0}{\ln 2} \cdot \frac{T_{1/2}}{T_{acq}} \cdot \left(1 - e^{-\frac{T_{acq}}{T_{\frac{1}{2}}} \ln(2)} \right)$$

Here can be shown that, as we are going to use a time of acquisition $T_{acq} = 1\ min \ll T_{1/2}$, we can approximate by a Taylor series that $A_{ave} \approx A_0$. The problem here is related with the free-space disk. After the development of the other tests we were left with little space in disk, something that combined with the large exigencies the simulation at higher doses has (with intermediate files that may achieve sizes of terabyte orders) made impossible to do repeatedly computations. In fact, we could only retrieve one measure for a very low activity, $A_0 = 0.10\ mCi$, which was enough to implement the stablished protocol in the NEMA, but we will not be able to arrive to the final results as we need the distribution for different activities on the whole range. We will discuss it later on. In any case, the chosen parameters for the simulation are:

***Table 13.** Chosen parameters for the SF test.*

Parameter	SR Value
Divisions	6
Reconstruction type	FBP2D
Simulation time (s)	60
Total dose (mCi)	0.1

For each of the activities, we are asked to extract the matrix of sinograms of the simulated image and, for each of the individual sinograms, remove all the counts that are farther from 12 cm of the centre of the scanner. We can compute the pixels from which we have to set all values 0, by finding the size of the sinogram bins, which is simply:

$$|N_{pix}| > \left| \frac{n_{bins} \cdot 2}{d_s\ (cm)} \cdot 12\ cm \right|$$

Where d_s is the diameter of the scanner, which can be retrieved from the scanner parameters files. Now, we are required to align the sinogram counts so that for each projection (i.e., each sinogram row) of each slice. Note that, in contraposition of what we did in the Sensitivity test, now we are interested in grouping sinograms in slices, not just the count sums. Thus, we slightly modified the `sinogramator( )` function (see *Sensitivity* section for the pseudocode) by removing this last step.

Thus, the algorithm starts by searching the maximum value of the row. Next, it finds the distance between the index of the maximum and the central index. Finally, it moves the row to align all the maximums at the central index of the sinogram. An example of the procedure and our aligned sinograms can be found in *Figure 25*.

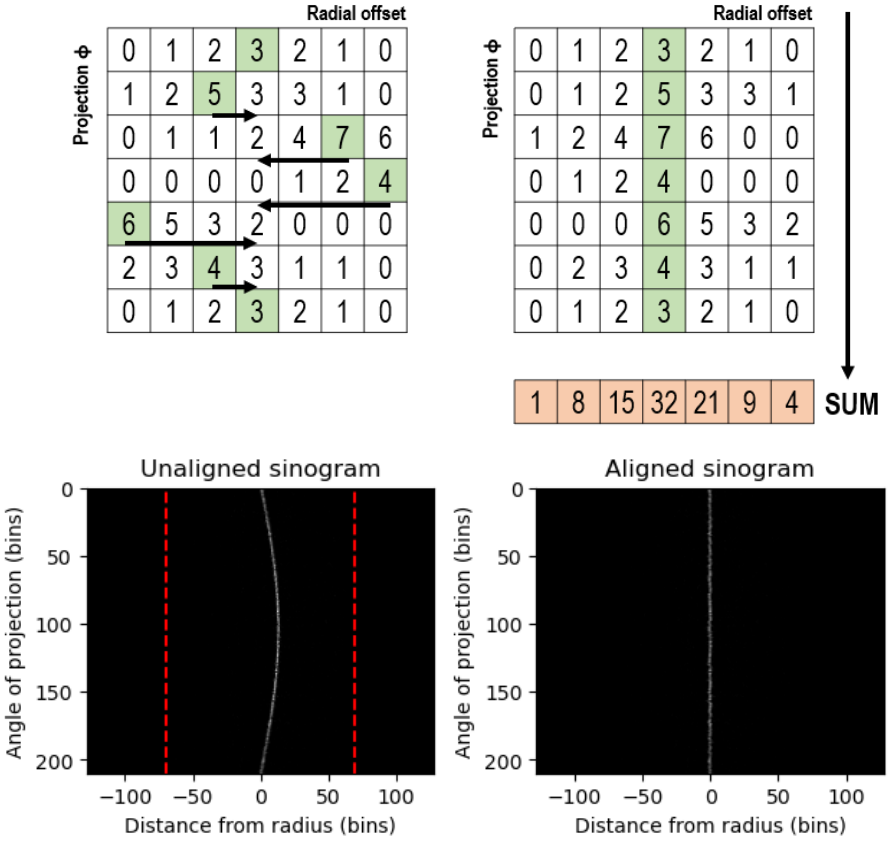

Figure 25. *Example of the alignment in a sinogram (top) and the resulting alignment for the SF image with the 12 cm bands in red (bottom).*

Thus, as shown above, we need to extract the projection summation (i.e., the sum of each of the columns) from which we can build a count profile that will be naturally centred at offset 0:

$$C(r) = \sum_{\phi} (r - r_{max}(\phi), \phi)$$

From it, we are asked to interpolate (similarly as in the SR test) the count values at the ±20 mm position (noted as CL and CR, respectively). The area under the intersection between the profile and the straight formed by the two interpolated points will correspond to the scattered counts and the randoms (C_{r+s}), while the others are assumed to be the unscattered counts (true events, C_t). Then, the total counts (C_{TOT}) can be simply taken as the sum of the projection. In *Figure 26* is attached a scheme of what would be expected in an arbitrary simulation:

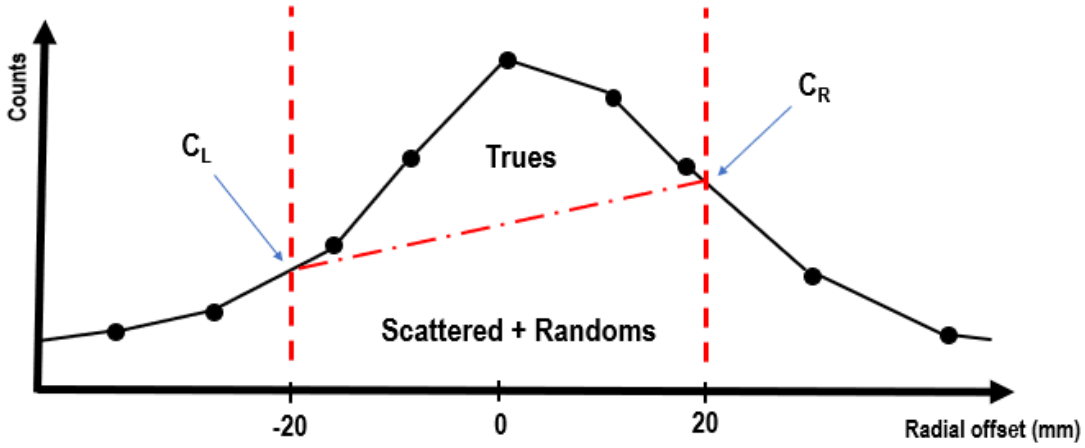

***Figure 26**. Scheme of the splitting between scattered and unscattered photons.*

Hence, the number of scattered + randoms counts can be approximated as:

$$C_{r+s} = \frac{C_L + C_R}{2} \cdot 40\ mm \cdot p_{size}$$

where p_{size} is the bin size of the sinogram. Finally, the counts of the random events are the sum of the random sinogram (which can be obtained directly from SimPET) after removing the counts farther than 12 cm, as we did at the beginning. From these measurements, we can extract the count rate profiles of the total, trues, scattered and randoms events per slice, respectively:

$$R_{TOT,i} = \frac{C_{TOT,i}}{T_{acq}}; \quad R_{t,i} = \frac{C_{TOT,i} - C_{r+s,i}}{T_{acq}}; \quad R_{r,i} = \frac{C_{r,i}}{T_{acq}}; \quad R_{s,i} = \frac{C_{r+s,i} - C_{r,i}}{T_{acq}}$$

Additionally, we are also asked to compute the Noise Equivalent Count Rate (R_{NEC}), which is a figure of merit related with the signal-to-noise ratio after corrections in the scanner are applied:

$$R_{NEC,i} = \frac{R_{t,i}^2}{R_{TOT,i}}$$

The total count rates of each type of event are set to be the sum over all the slices, that is simply, $R = \sum_i R_i$. With all these measurement we can finally compute the scatter fraction (for each slice and for the total system, respectively):

$$SF_i = \frac{C_{r+s,i} - C_{r,i}}{C_{TOT,i} - C_{r,i}}; \qquad SF = \frac{\sum_i C_{r+s,i} - \sum_i C_{r,i}}{\sum_i C_{TOT,i} - \sum_i C_{r,i}}$$

What we are required next, is to compute all these magnitudes for increasing activity concentration values, and extract the activity at which the peak for the NEC and trues appears. Besides, the SF at the NEC peak is also requested. However, as we justified before, we do not have enough computational resources to compute the measures, so we leave as future work to repeat them and extract their maximum (which is trivial).

Instead, as an alternative process of validation, we decided to take profit of the SimPET features and compute the SF and the count rates directly from the individual sinograms (i.e., the trues, randoms and scatter sinograms), which is much simpler than the procedure stipulated in the NEMA as we simply need to load these and sum them for each slice. In the *Results* section we will compare both metodologies.

6. Results and Discussion

In this section we are presenting the results obtained after the simulations and implementations applied in the Detailed Engineering. Furthermore, a discussion on the validity and consequences of these values is also included. As we already mentioned, there is a lack of bibliography on the topic and consequently there are few options to do comparatives with our values, so these have to be considered carefully.

6.1. Spatial Resolution

The values of the resolutions computed and their comparison with the reference papers [28, 29] we are using in this project are summarised in the following table:

Table 14. *Resolution values (FWHM and FWTM) for the SR test.*

Measure	Radius (cm)	FWHM (mm)	Relative difference	FWTM (mm)	Relative difference
Transverse	1	5.82	-7.3 %	12.77	10.6 %
Axial		3.95	-13.3 %	8.80	-26.3 %
Radial	10	5.35	-22.9 %	9.98	-27.1 %
Tangential		5.79	-15.1 %	10.87	-9.8 %
Axial		4.05	-33.7 %	9.21	-20.0 %

From the table above, we can see that in most of the cases we are obtaining values under the ones retrieved in the real experiment. First of all must be stated, that according to the developers of SimSET, a margin between 10-15% between real and simulated values may be expectable, as it is not feasible to obtain exactly the same results as in the experimental set. Besides, even in real

tests there is a certain variability due to changeable conditions. We can see thus, that many of the retrieved measures lay under or close to this interval.

However, there are some measures that seem to be especially far from the real values, particularly the radial and axial resolutions for R=10 cm. As the NEMA tests require, these measures have to be taken into the best possible conditions, for example, by removing the smoothing filters (which reduce the noise but degrade the resolution of the image). In real environment, however, there might interfere several conditions that are not considered in the simulation, such as the presence of noise or unidealities in the detectors. Consequently, these simulated results could be taken as the ”more optimistic results possible” and as a lower limit of resolution for the real environment.

On the other hand, the processing of the data may have also played a role. The interpolation of the data is performed under the linear relation, which in some cases may result in a good approximation, especially for distributions with a high density of measures, which is not the case. Thus, these approximation error may also contribute to inaccuracies on the results. In any case, the NEMA premisses have to be strictly followed.

We can also have a look at some of the resolution profiles obtained. In *Figure 27* it can be seen as an example the source point (0, 10) cm at the centre of the FOV.

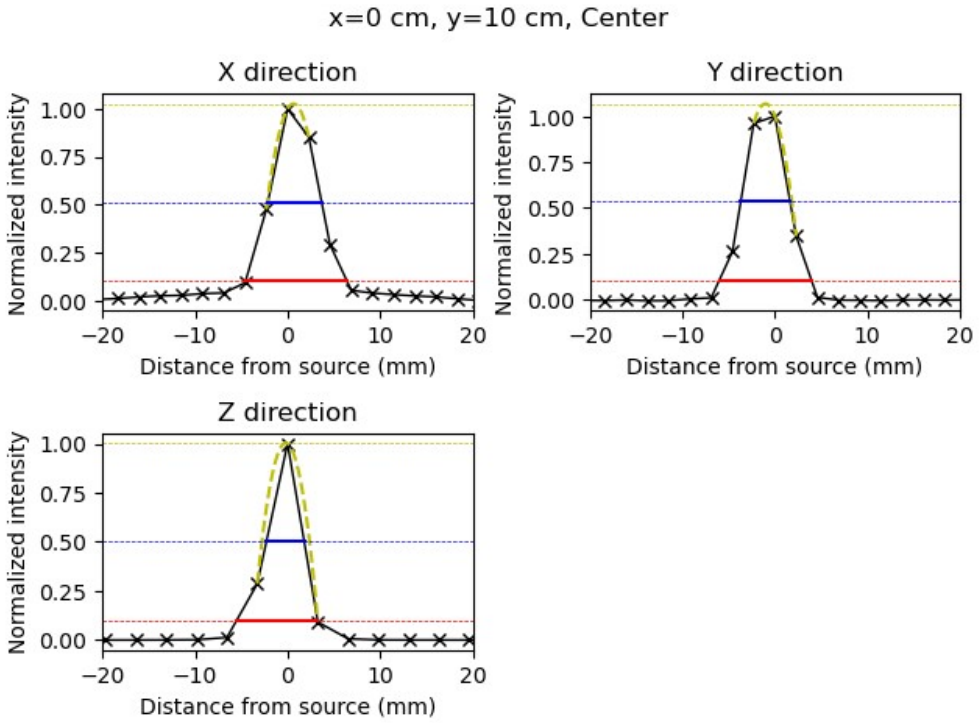

Figure 27. *Intensity profiles for the x=0cm, y=10cm source in the centre of the FOV.*

In these plots the lack of data, which may lead to a poor interpolation, gets clear. However, the amount of data is limited by the binning of the scanner (i.e., the size of the reconstructed image pixels), so there is nothing that can be much done in that sense. Also remarkable is the lack of noise in the simulation, as the baseline rapidly moves to zero before and after the peak. Under noisy images, the presence of spikes or a stepped baseline might be possible.

If we have a look at another set of profiles, e.g., for the source (10, 0) cm offside the FOV, shown in *Figure 28*, can be seen that similar profiles are obtained for the three directions. These profiles are actually similar at visual inspection for all the other positions, and more crucially, they remain similar after new simulations. In fact, the values of the FWHM and FWTM, and consequently the resolution values, also keep relatively constant after different executions, suggesting the stability and robustness of the results.

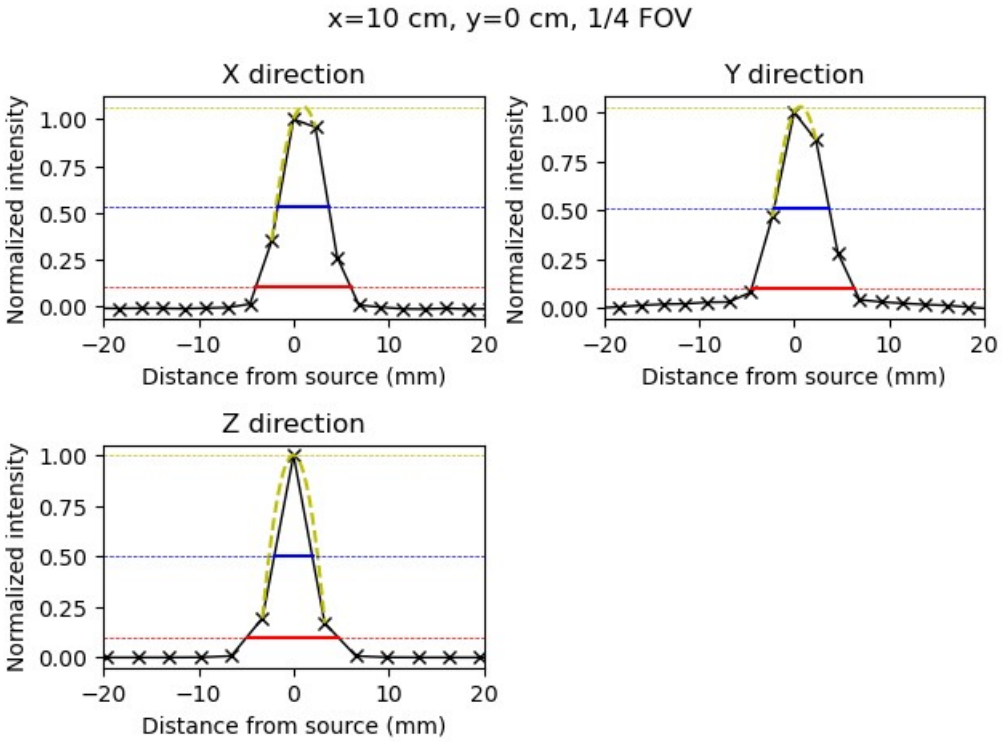

***Figure 28**. Intensity profiles for the x=10cm, y=0cm point source at 1/4 of the FOV.*

6.2. Sensitivity

After we executed the simulation for the five sleeves we obtained the following adjusts for the count rates:

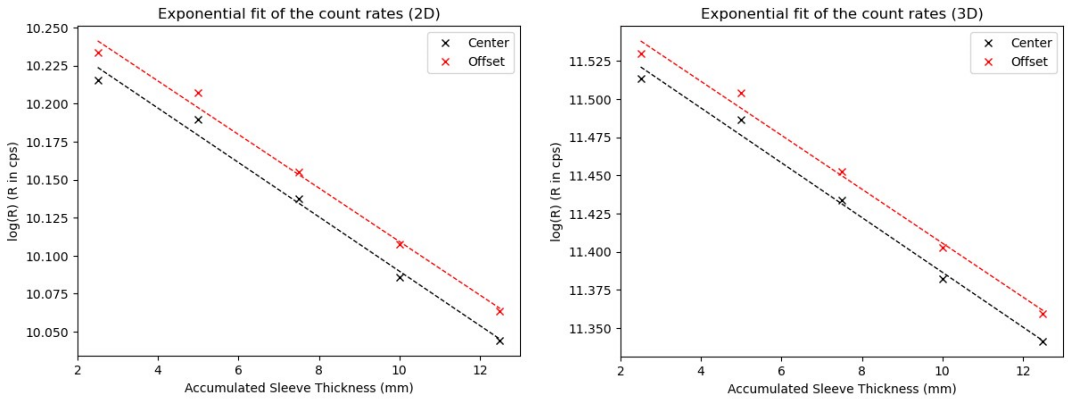

***Figure 29**. Exponential fits of the count rates for the five sleeves.*

From these, we could extrapolate the count rate values at 0 thickness (R_0), as stipulated in the *Detailed Engineering* Section, and by normalizing them to the activity used we could retrieve the total sensitivity of the system. The following table collects all this data, together with the sensitivity reference values:

Table 15. *Sensitivity values (theoretical and experimental) for the different positions and reconstructions.*

Measure	R_0 (cps)	Total sensitivity (cps/kBq)	Theorical sensitivity (cps/kBq)
2D and Centre	28733	15.53	1.92
2D and Offset	29210	15.78	1.95
3D and Centre	105175	56.85	8.99
3D and Offset	106881	57.78	9.26

In *Table 15* can be observed that the sensitivity values obtained are significantly far away from the ones obtained in the experimental setup. At first sight, it may be thought that the measures do not have sense at all, but if we look at the ratios between the 3D and 2D measures:

Table 16. *Ratios between the 3D and 2D reconstruction sensitivities.*

	Simulation ratio	Experimental ratio
Centre	3.7	4.7
Offset	3.7	4.7

We realise that the ratios between simulation and experimental measures are reasonably close, suggesting that there might be some kind of factor acting within the simulation and linearly increasing the results. Further work in this aspect should be required. In any case, from the previous tables it can also be noticed that the results between centre and offset are relatively close, showing an stability of the sensitivity independently of the considered zone in the scanner. In other words, we are finding here that the capacity of detecting photons is maintained constant in all the FOV zone, something that is desirable in a PET scanner in order to avoid inhomogeneities in the acquisition.

On the other hand, there is also another similarity of the simulated results compared to the experimental ones. In both cases, we find that the sensitivity in the 3D reconstruction is clearly higher than in the 2D case. This is expectable, because as we previously justified, in 2D we are limiting the acceptance of photons in between a certain number of rings, while 3D can be considered as a particular case where we accept photons between the maximum ring difference possible. Thus, it is logical to think that, as we are considering more ring pairs for the simulation, the capacity of photon detection may also be higher.

Finally, the NEMA tests also require to plot the sensitivity profiles for the first sleeve at the centre of the FOV:

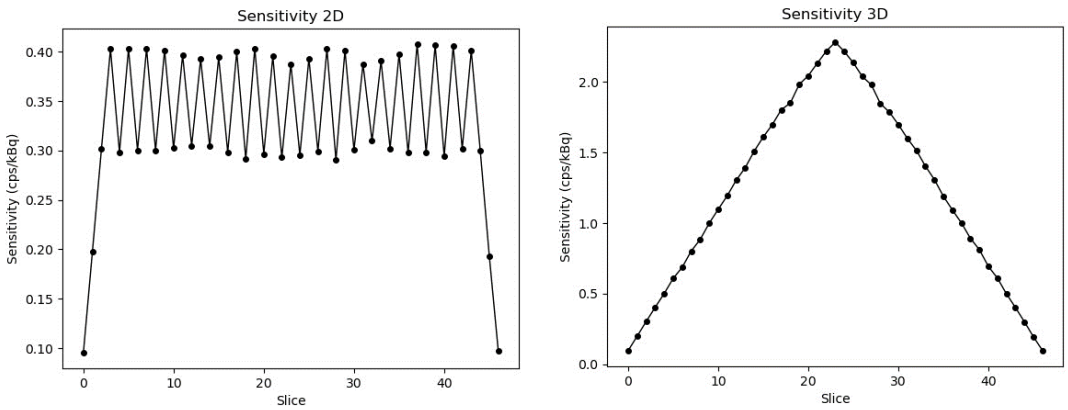

Figure 30. *Sensitivity profiles for the 2D (left) and 3D (right) cases.*

Although the values are logically displaced (as they depend on the S_{tot}), the shape of both distributions seems to be in accordance of what would be expectable for both reconstruction methods. For the 2D, we find these spikes caused by the alternance on the number of summed sinograms, while in the 3D case, this characteristic triangular form appears as a progressive sum of sinograms until reaching the central slice.

6.3. Image Quality

So after drawing the 12 ROIs at the 5 selected slices, we retrieved the recovery coefficients of each of the spheres and the background variability:

Table 17. *Recovery coefficients and background variability for each sphere.*

Diameter (mm)	QH (%)	Background variability (%)
10	158.8	11.6
13	196.4	8.6
17	211.7	7.3
22	230.2	6.0
28	222.6	5.0
37	235.6	4.3

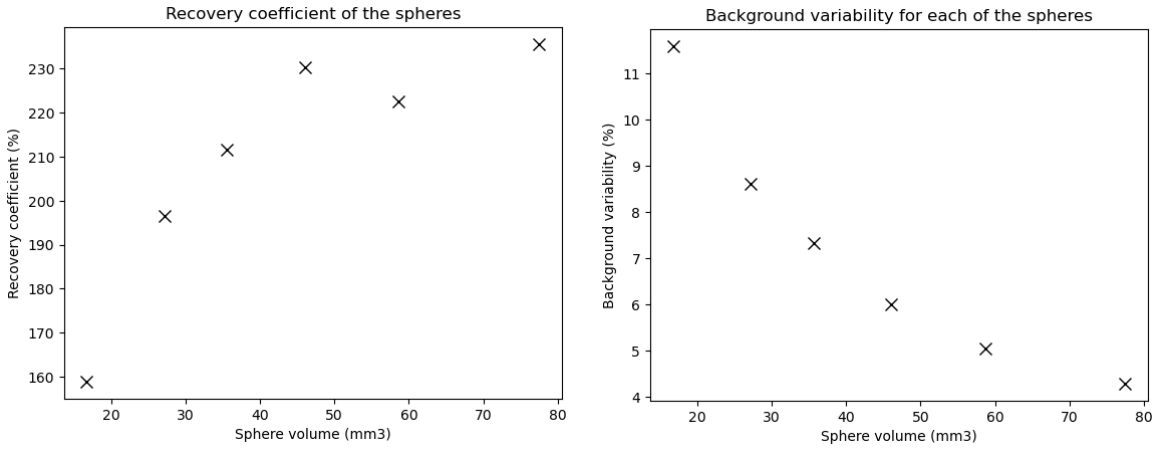

Figure 31. *Profiles of the recovery coefficients and the background variability.*

We have to note that, as we are using an example image and not the simulated data, we do not have any reference to contrast our data, meaning that we cannot ensure our values are in accordance with the real experimental values. However, we can study the behaviour of the curves, which are actually consistent with the expected one.

On the one hand, the recovery coefficient tends to increase as the sphere volume does, as we observed how the central intensities on the spheres seem to be higher for larger areas in the slice; however, this effect is mitigated when the volumes of the spheres get even larger values, forming this kind of plateau in the three largest spheres. On the other hand, the background variability tends to decrease with the sphere volumes. This is also in accordance with statistics, as we defined this variability as a function of the standard deviation of the background ROIs. Hence, note that for larger spheres, we are averaging a larger population and consequently we could expect to reduce this variance, which effectively happens.

Finally, we can also apport the values for the ROIs in the lung insert, as asked:

***Table 18**. Residual error in the lung insert for each slice.*

Slice	88	98	108	118	128
ΔC_{lung} (%)	8.6	9.8	7.3	9.0	6.7
Average ΔC_{lung} (%)	8.3				

The results seem to remain reasonably stable among the different considered slices, something that is desired in PET imaging as otherwise will mean there are some unidealities in the scanner that lead to the introduction of artifacts and noise. In any case, as this value corresponds to the residual error after the corrections of the image, ideally we are interested in minimizing ΔC_{lung}. In our image, we obtained a residual error of 8.3%, which seem to be low enough to be considered acceptable (for example, compared with other paper values, where it can arrive up to 25%.

These values were taken for a ratio of activities concentration between spheres and background of L/B=4. As the NEMA directrices require, it would be required to repeat these measurements for L/B=8, but as we do not have at our disposal that image, we will leave its computations, together with using this implemented workflow for the simulated data, as further work to develop.

6.4. Scatter Fraction

As we already commented in *Detailed Engineering*, due to time but especially resources limitations it was not possible to repeatedly retrieve the measures concerning this test for different activities, as it is required by the NEMA protocol. Indeed, the lack of free-disk capacity has arised to be one of the main limitations of the project, as one point data represent 10GB of capacity, but more crucially, the intermediate files might demand even higher capacities. For reference, in experimental papers are usually taken about 20 measures, which exceeded by much the free capacity of the working computer. One solution could be deleting 'at flight' the generated files while keeping the data, but this escaped the temporal scope of the project and is left as future work.

In any case, we reached to obtain results for one activity with a total dose of 0.10 mCi. Therefore, we will present the results obtained by following the NEMA directrices, and for completeness we will validate them by directly computing the Scatter Fraction from the splitted sinograms (i.e., trues, randoms and scatter) profiting from this feature of SimPET. So first of all, we proceeded to alineate the sinograms, as we explained in the *Detailed Engineering* section. Next, after the summation along each angle projection with the same radius offset (that is, each column of the aligned plot) we could do the adjust to split the scattered and unscattered photons per each slice:

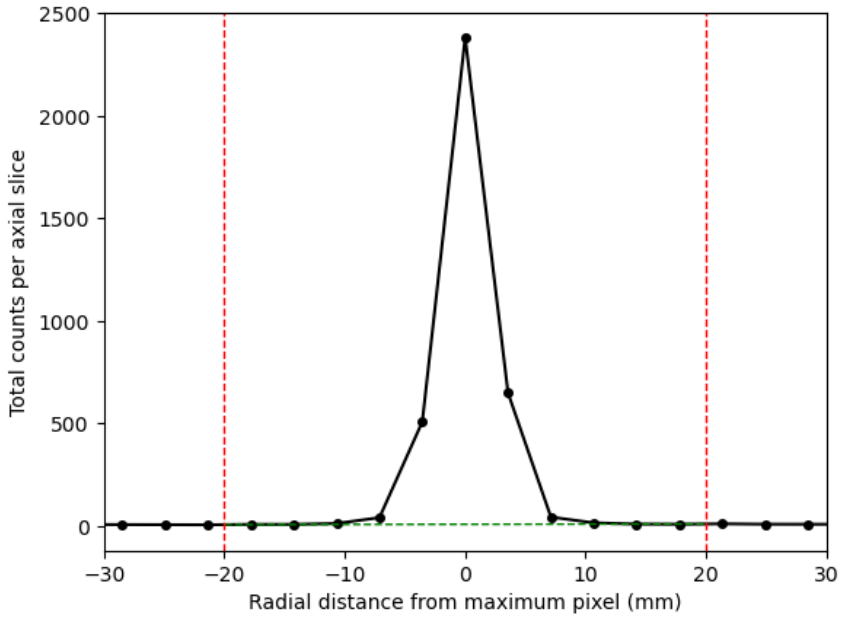

***Figure 32**. Profile of the total counts in one slice in function of the radial distance offset, with the adjust (green) between the 40 mm band (red).*

If we compare the plot obtained above, compared with the expected shape we described in *Detailed Engineering* we see that the width of the peak is too narrow, meaning that at the time the profile arrives at the ±20mm strip, the number of counts are too low and almost located in the baseline. Thus, when we compute the fraction of scattered+random counts, we get a tiny, almost negligible number that has no significance at all.

This fact generates an alteration of the results presented in *Table 19* on the system event rates, which consequently implies an unrealistic value for the scatter fraction of the system. For this reason, we proceeded to retrieve the values by directly by computing the number of counts on the splitted trues, scattered and randoms sinograms. Following we present the slice profiles computed with the NEMA procedure and by direct computation, for both 2D and 3D reconstructions:

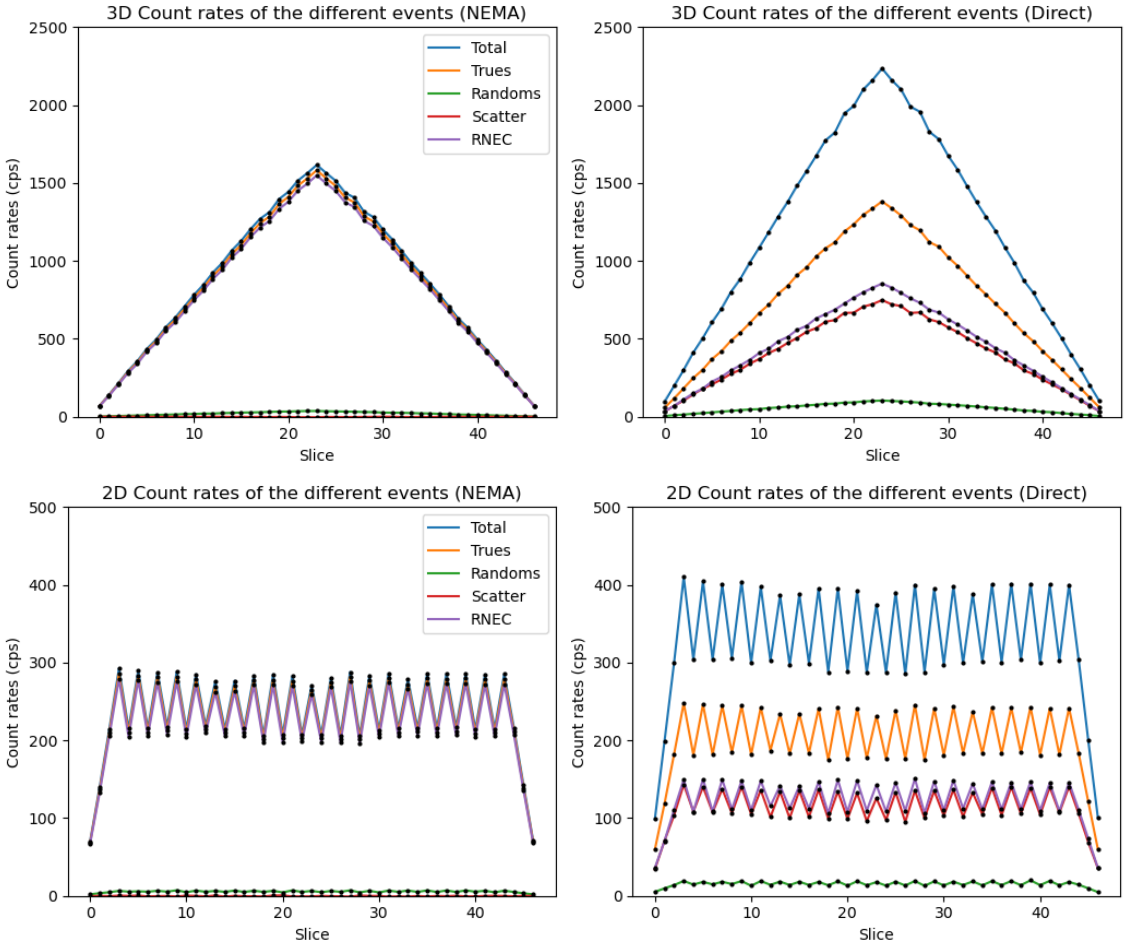

***Figure 33**. Comparative between the count rate profiles for the 3D (top) and 2D (bottom) reconstructions.*

With the different values of the events collected in the following table:

***Table 19**. Comparative of the total count rate event values for the 2D and 3D reconstruction methods.*

	2D		3D	
Event	**NEMA**	**Direct**	**NEMA**	**Direct**
Total (cps)	11071	15465	40343	56128
Trues (cps)	10822	9405	39450	34368
Scatter (cps)	-11	5334	-43	19143
Randoms (cps)	260	725	936	2617
R_{NEC} (cps)	10578	5721	38577	21045

From the profiles on *Figure 33* can be seen, that this low profile of scattered photons obtained by the NEMA procedure, for both 2D and 3D, combined with a low profile of randoms, implies that the trues profile (and therefore, the R_{NEC}) is very similar to the total profile. In fact, the numbers obtained for the total count rate and the total trues rate of the system are also equiparable. Actually, we obtain that the scatter values are negative, something that does not have any sense as the count rate is a positive-defined magnitude, and consequently these results should be discarded. Due to this anomaly, the scatter fraction values and profiles also have not any solid basis:

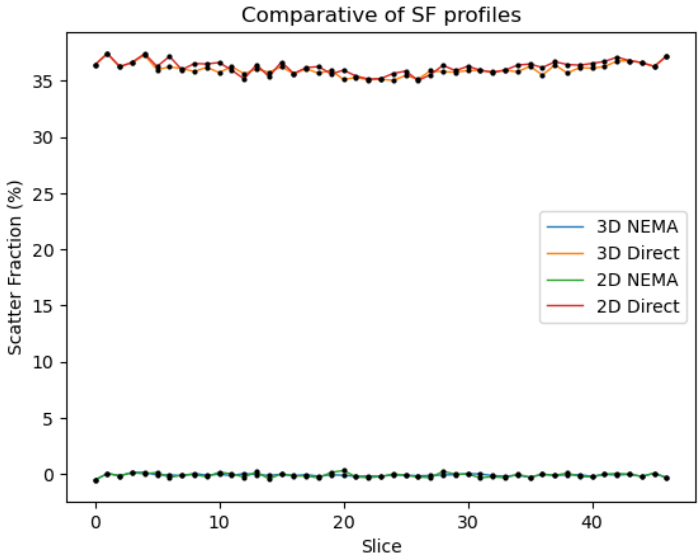

Figure 34. *2D and 3D Scatter Fraction profiles for the both methods considered.*

Again, we find high differences between the computations with the NEMA directrices and the direct computation. The values of the SF from the NEMA should not be taken into consideration, as these low-quality values of the scattered profiles lead to inconsistent total SF values (as before, negative values have no sense as the SF is a positive-defined magnitude). More surprisingly, we can notice that comparing 2D with 3D, either the values and the profiles are reasonably similar, something unexpected as, by broadening the range of accepted photons through more distant rings, more scattered photons should be detected in the tridimensional case.

Table 20. *Total system Scatter Fraction values for 2D and 3D reconstructions.*

	NEMA	Direct
2D	-0.11 %	36.19 %
3D	-0.11 %	35.77 %

Taking all this information into account, it is clear that these results have no physical sense and could not be useful to validate the scanner. In fact, these inconsistencies may be due to the extremely low total dose value imposed in the simulation. As we presented, we used 0.10 mCi, which is equivalent (for a phantom of a volume of 22000 mL) to an activity concentration of 0.17 kBq/mL. This quantity is almost depreciable compared to the ones used in the reference paper, which arrive up to 70 kBq/mL. That might be the explanation of this lowered profile of scattered photons, which triggered the rest of illogical results. For higher activity concentrations, the scattered fraction is expected to be increased.

Returning to the whole test context, all these results form part only of one iteration of many that should be performed for increasing activity concentrations. From these, the profiles of the total trues and R_{NEC} count rates should be plotted, and the peaks extracted. However, as we did not have enough computational resources, we could only perform this first minimal-dose iteration, which can be taken as the first iteration, and considered as a noisy measure. Hence, the repetition of these measures for higher activities is left for further works.

7. Execution chronogram

In this section we are presenting one of the crucial parts on the planification of a project: the expected temporal distribution. Usually, time is one of the limiting factors of the projects, as in our case, so it is of high importance to correctly split the tasks and assign them a reasonable amount of time. One positive aspect of our project is that, as it is based on independent tests, it is simple to divide the tasks into smaller packages, which facilitates the planification. The total duration of the project was set to be not more than 4 months, between the beginning of February to early June, when the projects must be presented.

In *Figure 35* can be seen the WBS of the project. In the second Level are included the usual packages of any project (Management, Prior research and Presentation of the Results), while the particular packages of our project (i.e., the NEMA implementation) have been grouped by similarities between tests. An extra package for the integration of the code, together with exploring his applications, is also disposed.

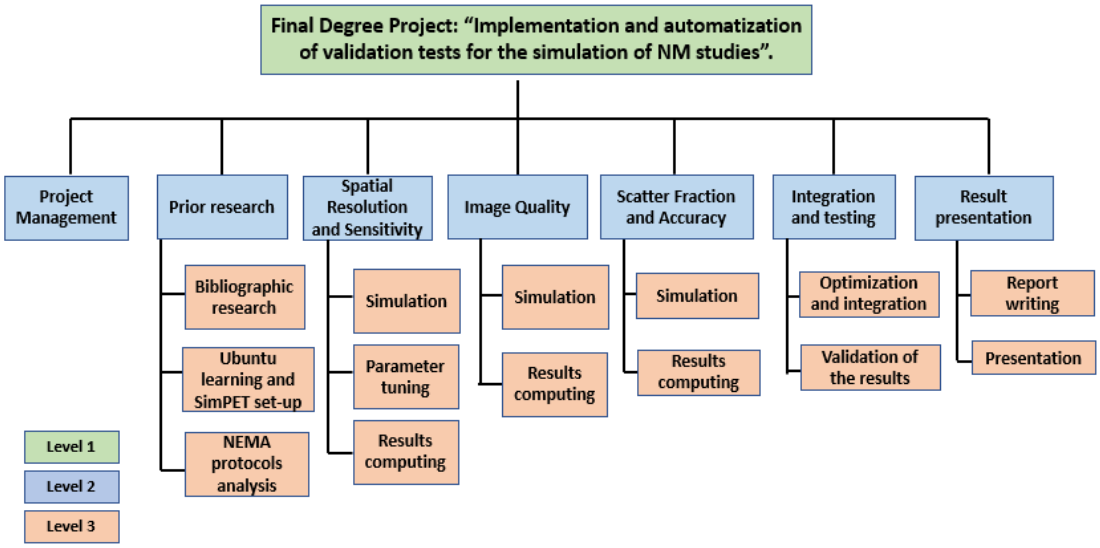

***Figure 35**. WBS chart of the project.*

From these, we can also assign each of them an expected time and the precedences (that is, the activities that need to be finished before starting another). For example, as we cannot afford to perform simulation on two test simultaneously, we will need to wait until one is finished to start the next. However, in order to save times, we can process the previous test data while the next is simulating. The following table shows the precedences and the assigned times:

Table 21. Precedence and duration of the WBS activities.

Code	Activity	Precedences	Duration (days)
A	Bibliographic research	-	21
B	Ubuntu learning and SimPET training	-	10
C	NEMA protocols analysis	A	12
D	Parameter tuning	B	15
E	SR and Sens simulation	D	12
F	Resolution and Sensitivity computing	C, E	10
G	IQ simulation	E	7
H	IQ computing	F, G	14
I	SF and Accuracy simulation	G	15
J	SF and Accuracy computing	H, I	12
K	Optimization and Integration	J	14
L	Validation of the results	J	10
M	Report writing	K, L	15
N	Presentation	M	7

Finally, we also show the GANTT chart associated to the project:

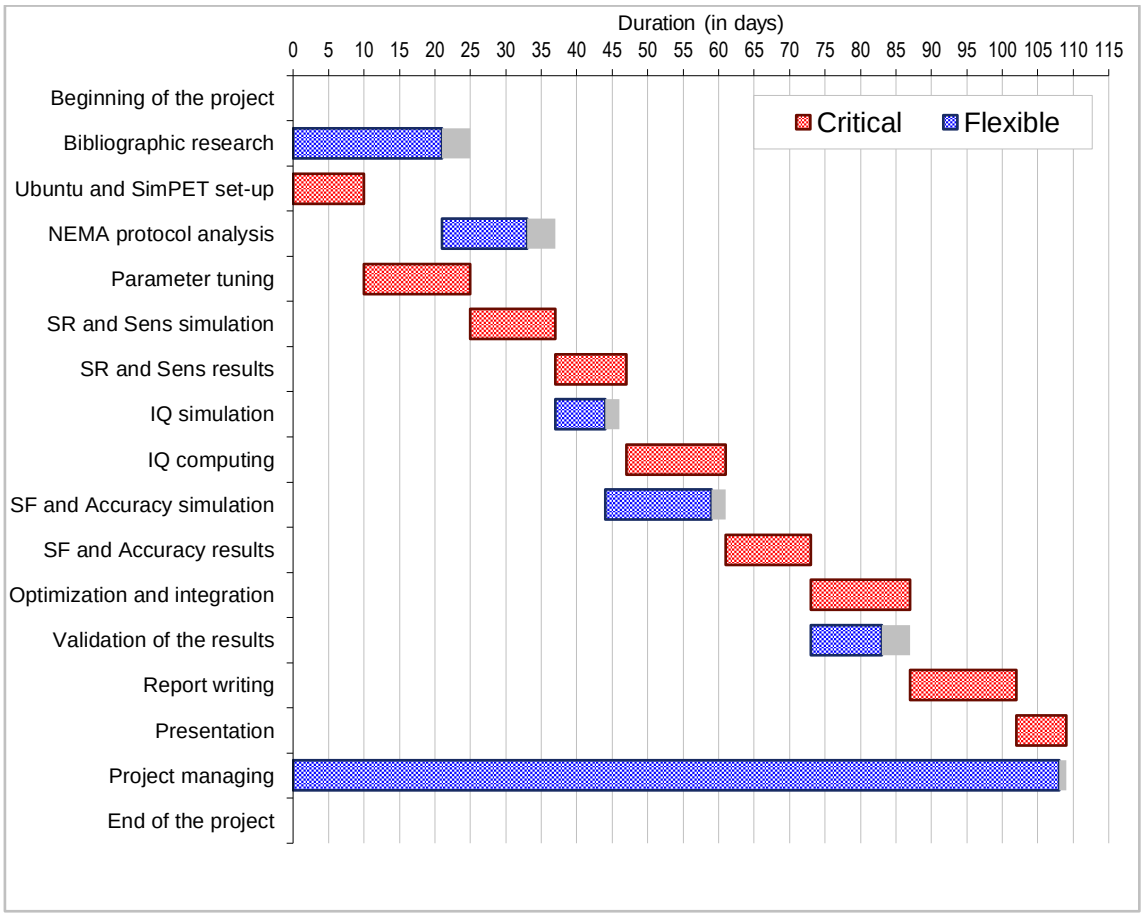

Figure 36. GANTT chart of the project.

The estimated time to execute the project is then 104 days, which lays below the 4 months limit. From the previous chart it can be seen, that the project is highly lineal in terms that most of the activities form part of the critical path, while the others show a very small margin of time. As stated in the precedence table, this project requires of the application of the steps in an almost sequential way, meaning that one step is highly dependent from the other. Thus, it is really important to avoid any kind of delay on the executions, otherwise the end of the project would be delayed as well.

Just to remark, it is noteworthy the significant amount of time devoted to prior work before the execution compared to other Final Degree Project. As justified during this report, it was crucial to perfectly know the working and premises of the NEMA, so a prior analysis of the documentation was of high importance.

8. Technical viability

In this section we will briefly study the technical viability of the project, by means of the SWOT analysis. This widely implemented technique inspects the strengths and weaknesses of a project (internal origin) but also its opportunities and threats (external origin). A list of these are collected in *Table 22*.

Table 22. *SWOT chart of the project.*

STRENGHTS	WEAKNESSES
1. High experience in programming, especially in Python, which has been used since long time ago. 2. Knowledge in Physics, which will help understand better the physics of the PET. 3. Accessibility to a multidisciplinary environment: the Nuclear Medicine service and the Image Computing Laboratory. 4. Support from both experts in Nuclear Medicine and the development team of SimPET.	1. SimPET still in development, which may bring some bugs in the program or inaccurate results. 2. Computers not extremely powerful and lack of free-disk space. 3. Small knowledge on Ubuntu environment and in script launching. 4. Low experience on community open-source project contribution and their associated workflows (e.g., use of Github).
OPPORTUNITIES	**THREATS**
1. Increasing interest in the clinical scope on quantification techniques and the development of associated tools. 2. Few simulation program developed, with a very reduced few of alternatives to our proposal. 3. Low budget project, without need of a large inversion to start it. 4. No need of retrieving data from patients, so there is no problem of data provision as it is created artificially.	1. Lack of bibliography and reference papers in the literature, which may difficult the validation of the results. 2. Lack of knowledge on the SimPET platform and small amount of documentation about the prior advancements on the project. 3. Poor developed NEMA protocols, which are sometimes difficult to clearly understand and to follow reliably their directrices. 4. Temporal limitation of the project, as the expected time for a Final Degree Project might not be enough as simulation requires extended amounts of time to be completed.

Having a look at the previous SWOT chart, we can see that there are many strengths and opportunities that suggest the project is feasible to be realised (at least, partially) given the resources at our reach and the previous knowledge on the topic. However, there are several weaknesses and threats that may difficult or slow down the project and that we have to neutralize.

On the one hand, it is noteworthy the threat of lack of information on references, the SimPET platform and the NEMA protocols. In general, the bibliographic research is a fundamental step on a project, especially on a scientific one, and without a strong basis the learning curve may result steeper and the further development may not be fast enough. In that sense, time may become then another threat if indeed there appear many difficulties to advance. To overcome these, we should also have in mind the support that will be provided from the Nuclear Medicine Service (for theoretical-related topics) and the SimPET development team (for more technical reasons). Another advantage in that sense is that there will not be a problem to obtain data as we will create it artificially, so that might be helpful to win some time in our favour. Besides, this fact may be taken as an opportunity as well, because if there is little development on the field, means that we can bring to the scientific community something innovative and new that might open a new page in the simulation studies.

On the other hand, the prior experience has to be analysed as well. On the one hand, the use of Python since the entrance in the University and the previously achieved Physics degree will avoid starting from zero and will represent a valuable help on speeding up the first steps of the project. On the other hand, there are many other aspects (like the use of Ubuntu or the development of open-source projects) in which we will need to start from scratch. Consequently, a previous phase of learning will be required before starting with the development, but it will be shortened due to some previous experience. Taking all of this into account, we can expect that the strengths of the project will help overcoming its weaknesses, and although there might be some threats, the potential opportunities in having success on the project have more weight. Consequently, we can consider that the project is feasible and possible to execute in the terms it has been raised.

9. Economic viability

In this section we will briefly study the economic viability of the project. As will be seen, as a programming project not a huge investment is required to start working on it, especially when considering that the used project is open-source and does not require any kind of data obtention from the patients directly. All the costs provided are approximate. On the one hand we must consider the human resources costs of the author of this project but also its advisors (we will assume a total hours in accordance with the stipulated in the Final Degree Project guidelines):

***Table 23**. Human Resources costs of the project.*

Concept	Unitary cost	Total cost
Arnau Farré	8€/h	2400 €
Aida Niñerola	13€/h	3900 €
Pablo Aguiar	13€/h	3900 €
Jesús Silva	13€/h	3900 €
Total Human Resources cost		14100 €

On the other hand, we have to consider the computational resources that we were provided:

Table 24. Computational costs of the project.

Concept	Unitary cost	Total cost
PC ('Gaming type')	1500 €	1500 €
2 Monitors	150 €	300 €
Laptop 'Gaming type'	750 €	750 €
Total Computational Resources cost		2550 €

Finally, we also have to consider our stay at the lab and the work at home, and the associated costs it brought during these months in terms of energy (assuming that every day we worked at home and only three days per week at the laboratory)

Table 25. Energy costs of the project.

Concept	Unitary cost	Total cost
Electricity costs	0.50€/day	54 €
Laboratory consumption	30€/day	1388 €
Energy total costs		1442 €

Thus, in sight of the previous values, we can assign an estimated total cost of the project around 18000 €. Compared with other projects, this does not represent an extremely high inversion (being most of the budget being paid in salaries, that is monthly). Besides, the University of Barcelona will be on charge of these costs, excepting the salaries of the directors of this work, in which case the University of Santiago de Compostela is responsible. Having all of that in mind can we assume that this project is economically viable.

10. Legal aspects and Normative

Generally speaking, all the Final Degree Projects are subject to the Integrity Code in Research for the Barcelona University. This text is governed by the principles of honesty, responsibility, reliability, rigor, respect and independence, and appeals all the academic and research staff of the entity. It fundaments the concept of intellectual authorship and rejects fraudulent circumstances like plagiarism, falsification and fabrication of data. In that sense, as the developers of SimPET are adscripted to the Santiago de Compostela University, this project is also subject under their Integrity Code.

On the other hand, we have to remember that SimPET is an open-source code, meaning that it is freely available and can be redistributed and modified by any programmer, naturally with an adequate citation. Consequently, the material derived from this work, when incorporated to the main program, will also be freely available. In that sense, the already done and the future works are subject to the Contributor Covenant Code of Conduct, which sets the rules for a correctly-working collaborative community. Besides, both SimSET and SimPET are licensed under the

General Public License, which allows anyone to use and modify the code, but under the condition that derived softwares should also be licensed under the GPL.

Concerning the NEMA organization, as the protocol itself specifies, it is discharged of any possible safety or health related issues as their responsibility only relies on certifying. NEMA also has no responsibility on the individual use of the information of their protocols.

Clinically speaking, it is fundamental to resemble that this software cannot be used for clinical purposes (i.e., as a support for diagnostics) until it receives the corresponding CE certification. Until then, this software shall only be used as a research tool and for the development of this or other tools.

Finally, it is interesting to note that by the introduction of simulating studies we are removing the legal aspects concerning the patient, as for the implementation of the NEMA tests all the maps and therefore the final images are artificially generated, meaning that no personal data is treated and we are not under any privacy policy. This significantly simplifies the handling of the data.

11. Conclusions and future lines

11.1. General conclusions

To sum up, we can say that returning to the initial objectives we stablished, we have partially accomplished them. On the one hand, after a deep and intensive study of the NEMA protocols, their directrices and how are they executed in real clinical practice, we can assume that we reached a reasonable high degree of understanding from them, something critical as they need to be followed very reliably. This led us to implement four of the five tests, as by time limitations it was not possible to work with the last of them (Accuracy). As a consequence, the code could also not have been fully integrated yet, although the tests have been already individually prepared to be merged in a generic code. Naturally, the subsequent steps of using the results for validating new scans could not been accomplished too, as not all the tests were finished.

As we already commented in previous sections, there have been three limiting factors, being the main of them the temporal availability. The expected times showed Execution chronogram dilated since the beginning as the learning curve of SimPET was steeper than expected. This early delay generated a cascading effect that dilated the execution times of the subsequent times. This fact resembles the crucial importance of a good time planification at the start of the project.

But another factor that at the same time played a role on delaying the obtention of results was the lack of computational resources. Again, the needs we expected at the beginning were clearly not enough with the selected working computer, as there was not enough space to deal with all that amount of results and that difficulted much the process.

The validation of the results was not also a trivial task, as the lack of bibliography on the topic did not work. In the *Results and Discussion* section we compare some of the values with a Reference paper, but it must be said that there were a little range alternatives to choose and the validity of

their results, as an experimental set-up, should be carefully considered. In any case, as said, we obtained some values for each of the tests.

Concerning Spatial Resolution, we worked with the premise of considering acceptable a deviation of about a 15% compared to the experimental results. In that sense, half of the measures lay in this range, but the other half might not be considered acceptable, so there is still work to do in terms of adjusting the simulating and scanner parameters to reach an acceptable range. Similarly for the Sensitivity test, the profiles seem to be in accordance with the expected behaviour but further work on adjusting the total sensitivities of the system needs to be done. For the Image Quality test, we already justified the absence of reference papers to compare our data with, but at least the obtained values seem to be in accordance with the theory, especially the background variability, which tends to decrease for the higher spheres, and the lung insert, showing lower values as desired in an IQ test. In terms of the Scatter Fraction test, we cannot extract many conclusions on the results themselves as they have clearly no physical sense, but we already shown that the method suggested in the NEMA does not work for very low activity values.

In any case, as a general conclusion on this project could be said, that we have shown that the implementation of the tests is possible and feasible, if the resources and time allow it. As there are only few trials on implementing the NEMA tests in the literature, it was not very clear at the start of this project if it would be possible to implement all these tests computationally. Although for different reasons we did not reach a fully implementation on some of them, we have showed that it is not only possible to implement them in the SimPET environment, but also to automatize them to give a set of values that allow the user to validate a scanner.

11.2. Future work

For all these reasons, there is much work left to do in the future, basically to fulfil the objectives that remained unaccomplished. The most immediate step is to finish implementing the Accuracy test, which may not bring many difficulties as its working principle is very similar to the Scatter Fraction test.

Next, it is required to find a computational alternative on the actual working computer, with capacity and power enough to handle with all the tests, especially with the IQ and the SF large files. Without them, it will not be possible to obtain significant results on them, as we already justified that we need to simulate large doses and times, which will generate these heavy files.

Following, the integration of the code to one single script would then be possible. As we commented in the conclusions, this step might not be problematic as the already implemented tests are thought to be in a generic script and coded in that sense.

Finally, when after checking that the single script works properly, the installation and addition to the main SimPET program will be necessary to enable its use to the SimPET users.

12. Bibliography

[1] López-González, F.J., et al (2020). Intensity normalization methods in brain FDG-PET quantification. NeuroImage (Orlando, Fla.), vol. 222, pp. 117229-117229.

[2] Carson, Daube-Witherspoon, M. E., & Herscovitch, P. (1998). Quantitative functional brain imaging with positron emission tomography. Academic Press.

[3] Paredes-Pacheco, José, et al. "SimPET-An Open Online Platform for the Monte Carlo Simulation of Realistic Brain PET Data. Validation for 18 F-FDG Scans." Medical Physics (Lancaster), vol. 48, no. 5, 2021, pp. 2482–93.

[4] NEMA Standards Publication NU (2007): "Performance Measurements of Positron Emission Tomographs". National Electrical Manufacturers Association.

[5] Edwards, C. L. (1979). "Tumor-localizing radionuclides in retrospect and prospect". Seminars in Nuclear Medicine. 9 (3): 186–9.

[6] Cicone, Francesco, and Philipp Lohmann. "Editorial: 'Nuclear Medicine in Cancer Diagnosis.'" Frontiers in Medicine, vol. 9, 2022, pp. 1035994–1035994.

[7] Carrió, Ignasi, and Pablo Ros. PET/MRI : Methodology and Clinical Applications. Springer, 2014.

[8] Cadoux, Franck, et al. "The 100µPET Project: A Small-Animal PET Scanner for Ultra-High Resolution Molecular Imaging with Monolithic Silicon Pixel Detectors." Nuclear Instruments & Methods in Physics Research. Section A, Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 1048, 2023, p. 167952–.

[9] Bailey, Dale L. Positron Emission Tomography: Basic Sciences. Springer, 2005.

[10] Wernick, Miles N., and John N. Aarsvold. Emission Tomography the Fundamentals of PET and SPECT. Elsevier Academic Press, 2004.

[11] Reynés-Llompart, Gabriel, et al. Image Quality Evaluation in a Modern PET System: Impact of New Reconstructions Methods and a Radiomics Approach. 2019.

[12] Khalil, Magdy M. Basic Science of PET Imaging. Springer International Publishing, 2017.

[13] Fahey, Frederic H. "Data Acquisition in PET Imaging." Journal of Nuclear Medicine Technology, vol. 30, no. 2, 2002, pp. 39–49.

[14] Ito, Hiroshi, et al. "Comparison of the Transient Equilibrium and Continuous Infusion Method for Quantitative PET Analysis of [C]Raclopride Binding." Journal of Cerebral Blood Flow and Metabolism, vol. 18, no. 9, 1998, pp. 941–50.

[15] SimPET Project | Accessible on: http://simpetweb.uma.es:8020/login | Retrieved at: 01 June 2023.

[16] NEMA Standards Publication (2001): “Performance Measurements of Positron Emission Tomographs”. National Electrical Manufacturers Association.

[17] Assié, Karine, et al. “Monte Carlo Simulation in PET and SPECT Instrumentation Using GATE.” Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 527, no. 1-2, 2004, pp. 180–89.

[18] Guerin, Bastein, and Georges El Fakhri. “Realistic PET Monte Carlo Simulation With Pixelated Block Detectors, Light Sharing, Random Coincidences and Dead-Time Modeling.” IEEE Transactions on Nuclear Science, vol. 55, no. 3, 2008, pp. 942–52.

[19] Salvadori, Julien, et al. “Monte Carlo Simulation of Digital Photon Counting PET.” EJNMMI Physics, vol. 7, no. 1, 2020, pp. 23–23.

[20] Pfaehler, Elisabeth, et al. “SMART (SiMulAtion and ReconsTruction) PET: An Efficient PET Simulation-Reconstruction Tool.” EJNMMI Physics, vol. 5, no. 1, 2018, pp. 16–18.

[21] España, S., et al. “PeneloPET, a Monte Carlo PET Simulation Tool Based on PENELOPE: Features and Validation.” Physics in Medicine & Biology, vol. 54, no. 6, 2009, pp. 1723–42.

[22] S. Ashrafinia, et al., “Generalized PSF modeling for optimized quantitative-task performance”, Phys. Med. Biol., vol. 62, pp. 5149-5179, 2017.

[23] SimSET Washington University | Accessible at: https://depts.washington.edu/simset/html/simset_main.html | Retrieved on: 03 June 2023.

[24] GitFlow Workflow | Accessible at: https://www.atlassian.com/es/git/tutorials/comparing-workflows/gitflow-workflow | Retrieved on: 29 May 2023.

[25] A. Ding, M. Mille, P. F. Caracappa, X. G. Xu, "Impact of body size of obese patients on PET/CT dose estimates: Monte Carlo calculations using a set of BMI-adjustable phantoms," 53rd Annual Meeting of the American Association of Physicists in Medicine, Vancouver, British Columbia, Canada, 2011.

[26] SimSET: Object Editor | Accessible at: https://depts.washington.edu/simset/html/user_guide/user_guide_index.html | Retrieved on: 10 May 2023.

[27] STIR User Manual. | Accessible at: https://stir.sourceforge.net/documentation/STIR-UsersGuide.pdf | Retrieved on: 14 May 2023.

[28] Bettinardi, Valentino, et al. “Performance Evaluation of the New Whole-Body PET/CT Scanner: Discovery ST.” European Journal of Nuclear Medicine and Molecular Imaging, vol. 31, no. 6, 2004, pp. 867–81.

[29] De Ponti, E., et al. “Performance Measurements for the PET/CT Discovery-600 Using NEMA NU 2-2007 Standards.” Medical Physics (Lancaster), vol. 38, no. 2, 2011, pp. 968–74.