“Monte Carlo code optimization for the application to Carbon ion Treatment Planning Systems”

Chiara D’Ambrosio

Supervisors

prof. Vincenzo Patera

prof. Maurizio Pelliccioni

Anno Accademico 2007/2008
Contents

Introduction 1

1 A BRIEF OVERVIEW OF HADRONTHERAPY 5

1.1 The cancer incidence and his treatment 5
1.2 The physical basis of Hadrontherapy 5

1.2.1 Lateral scattering and Range straggling 8

1.2.2 Beam fragmentation 10

1.3 Beam delivery systems 10

1.3.1 Passive shaping 11
1.3.2 Active scanning 13

1.4 The biological basis of ion beam therapy 15

1.4.1 LET, OER and RBE 15

1.4.2 Radiobiological rationale and patient selection for ion beam
therapy of cancer 18

1.5 The world hadrontherapy facilities and the Italian CNAO project 19

2 TREATMENT PLANNING SYSTEM FOR CARBON ION THERAPY 21

2.1 General description of a Treatment Planning System (TPS) for ion therapy 21

2.1.1 The patient modelling 22

2.1.2 Dose calculation algorithms 23

2.1.3 Radiobiological modelling for calculation of the isoeffective dose 24
2.2 The I.N.F.N. – TPS project .......................... 25
2.3 Use of a Monte Carlo code as a TPS validation tool .......................... 26
  2.3.1 The accurate target modelling .......................... 27
  2.3.2 The detailed fragmentation description .......................... 29
  2.3.3 The statistical accuracy in dose calculation .......................... 30
  2.3.4 Monte Carlo optimization tools development .......................... 32
2.4 The choice of the FLUKA Monte Carlo code .......................... 33
  2.4.1 FLUKA code capabilities .......................... 33

3 DELTA RAYS CONTRIBUTION ANALYSIS

  3.1 Delta rays contribution .......................... 35
  3.2 Dose deposition profile lateral broadening .......................... 37
  3.3 The FLUKA simulation physics settings .......................... 39
  3.4 The delta rays threshold effect on beam broadening .......................... 42
  3.5 CPU time improvement for different delta rays thresholds .......................... 46

4 TPS PARTICLE FLUENCE VERIFICATION AND OPTIMIZATION

  4.1 The mathematical bases .......................... 49
  4.2 The Monte Carlo fluctuations of $\mu_{i,\alpha}$ and $\sigma_{i,\alpha}$ .......................... 57
  4.3 The pencil beam definition .......................... 60
  4.4 The simulation geometry set up .......................... 62
  4.5 The TPS fluence verification .......................... 64
  4.6 The fluence optimization procedure .......................... 69
Conclusions and perspectives 77

References 81

Acknowledgements 87

Appendix

[A1] The world hadron therapy facilities 89
[A2] The include files: ptv.inc and fluence.inc 91
[A3] The fluence.f program 93
[A4] The fluence_lib.f program 103
INTRODUCTION

Cancer is the uncontrolled growth and spread of cells in the human body that, nowadays, represents the second leading cause of death in economically developed countries.

Radiotherapy, alone or in conjunction with surgery or chemotherapy, plays a leading role in the cure of tumors, especially if they are detected early. Best results are obtained when an optimized dose conformation to the target volume is achievable, that is with the use of a selective irradiation which spares the surrounding healthy tissues and organs at risk.

Radiotherapy treatments using light ions, such as $^{12}$C, come close to accomplish this highly desirable goal. This is due to the physical and biological properties of this kind of charged particles: a well localized absorbed dose distribution, thanks to the “inverse” depth dose profile with the Bragg maximum of ionization density at the end of the range and to the low angular scattering, and a large relative biological effectiveness which may be useful in the treatment of radio-resistant hypoxic tumor cells.

The described major advantages of hadrontherapy compared to conventional radiotherapy justify the worldwide fast growing of new proposals, designs and constructions of particle therapy facilities.

The possibility of routinely use carbon ions in the clinical practice of cancer therapy is obviously related to the availability of an accurate Treatment Planning System (TPS). This complex software, starting from the patient diagnostic images and the oncologist dose prescriptions, produces the set of directions, fluences and energies of all the ion beams which forms the radiation field and yields the optimized dose distribution.
Actually, the experience with carbon ions is quite limited due to the fact that only 3 facilities are worldwide in operation: HIMAC and HIBMC in Japan and GSI in Germany. Moreover only 2 analytical TPS are now commercially available: FOCUS/M by Mitsubishi Electric Co. and KonRad by Siemens. These treatment planning systems rely on analytical model to calculate the energy loss distribution of primary and secondary particles as a function of depth in water and make use of several simplifications. This approach is thus not always sufficiently precise because not all the physical interactions can be modelled and the target treatment volume description is based on a water equivalent approximation. Consequently, the TPS results have to be validated through experimental measurements of the depth dose profiles using phantoms, in order to verify the agreement between calculated and delivered dose distribution.

On the other hand, the TPS commissioning can also be performed using Monte Carlo simulations. Monte Carlo statistical methods are, indeed, the only instruments able to properly model the complex process of dose deposition due to the full 3D modelling of scattering, electro-magnetic and nuclear interactions and to an accurate description of the real composition of the human body, including also interfaces between tissues of different density and the presence of inhomogeneities. Unfortunately, Monte Carlo techniques are computationally intensive, that is are highly memory demanding and the CPU time needed to perform the detailed simulation of the treatment plan is, at the moment, too long for a direct application in clinical routine. Thus, it is necessary to develop new tools to optimize the efficiency of Monte Carlo simulations by a reduction of the overall processing time.
This work represents a contribution towards this aim, in the direction of the development of a reliable and accurate MC validation tool for carbon ion treatment planning systems by the use of the transport and interaction code FLUKA.

In the first chapter of this thesis a brief overview of the physical and biological basis of ion beam therapy is given, together with a glance at the world hadrontherapy facilities and the Italian CNAO (Centro Nazionale di Adroterapia Oncologica) project.

Chapter 2 presents the main characteristics of a Treatment Planning System for ion therapy and summarizes the motivations for the use of the FLUKA Monte Carlo code as a TPS validation tool, in the framework of the I.N.F.N. (Istituto Nazionale di Fisica Nucleare) TPS research project.

Chapter 3 and 4 are the description of the original work performed by the author for the Monte Carlo code optimization.

In the third the delta rays contribution to the dose deposition profile lateral broadening is analyzed and the delta rays production and transport threshold energy effect is presented together with the achieved CPU time improvement.

In the last chapter the developed tools for the verification and optimization of the TPS particle fluences are shown together with some preliminary results on a simple target configuration.
CHAPTER 1

A BRIEF OVERVIEW OF HADRONTHERAPY

1.1 The cancer incidence and his treatment

Nowadays, cancer is the second leading cause of death after cardiovascular diseases in economically developed countries [Ref. 1.1]. Estimates of the cancer incidence and mortality in the European Union (EU25) for 2006 have been recently published [Ref. 1.2] by the International Agency for Research on Cancer (IARC). Resulting estimated data are: 3.2 million incident cases of cancer diagnosed and 1.7 million cancer deaths and, unfortunately, the ageing of the European population will cause these numbers to continue to increase.

A significant fraction of cancers can be cured, especially if they are detected early when they are localized (before metastasis), by an adequate treatment like surgery, conventional radiotherapy (gamma and electrons) and chemotherapy. Further improvements in 5-year survival rates are also expected thanks to the diffusion of new radiotherapy techniques such as 3d-CRT (3-dim Conformal Radio Therapy), IMRT (Intensity Modulated Radiation Therapy) and Hadrontherapy, which allows an excellent dose conformation to the target volume (PTV, Planning Target Volume) with a resulting sparing of healthy tissues and OAR (Organs At Risk).

1.2 The physical basis of Hadrontherapy

Hadrontherapy is a particular type of oncological radiotherapy making use, instead of conventional photon beams, of fast non-elementary particles: protons,
neutrons and light ions such as Helium (\(^4\)He), Lithium (\(^6\)Li), Boron (\(^{10}\)B), Carbon (\(^{12}\)C) up to Neon (\(^{20}\)Ne). The use of heavier ions, as Silicon or Argon, is discouraged due to the fact that the amount of fragments produced in the target tissues by nuclear reactions increases with increasing mass and charge of the primary particle, thus deteriorating the physical selectivity of the beam.

In the energy range of therapeutically relevance, the process of ion energy loss is dominated by Coulomb inelastic collisions with target atomic electrons and the average energy loss per unit path length \(\frac{dE}{dx}\), also termed electronic stopping power, is described by the Bethe-Bloch formula [Ref. 1.3, 1.4]:

\[
\begin{align*}
\frac{dE}{dx} &= 2\pi N_A r_e^2 m_e c^2 \rho \frac{Z}{A} \frac{z^2}{\beta^2} \left[ \ln \left( \frac{2 m_e r_e^2 v^2 W_{\text{MAX}}}{I^2} \right) - 2 \beta^2 - \delta - 2 \frac{C}{Z} \right] \\
\end{align*}
\]

where \(N_A\) = Avogadro’s number, \(r_e\) and \(m_e\) = classical electron radius and electron mass, \(\rho\) = density of absorbing material, \(Z\) and \(A\) = atomic number and weight of absorbing material, \(z\) = charge of incident particle, \(v = \beta c\) = velocity of the incident particle, \(W_{\text{MAX}}\) = maximum energy transfer in a single collision, \(I\) = mean excitation potential, \(\delta\) and \(C\) = density and shell corrections [Ref. 1.5].

The latter expression allows to calculate, for a given initial beam energy \(E_0\), the so called mean range \(R = \int_{E_0}^{E_0} \left( \frac{dE}{dx} \right)^{-1} dE\), which gives a very close approximation to the length of the average path traveled by the ion in the material.

Robert R. Wilson, in a paper published in 1946 [Ref. 1.6], was the first to suggest that accelerated protons and light ions could be an effective cancer treatment method due to their “inverse” PDD (Percentage Depth Dose) curve: the dose increases with penetration depth and with decreasing particle velocity, giving rise to a sharp maximum (the Bragg peak) just before the end of the range.
The increase in ionization density towards the end of the particle track allows the treatment of deep seated tumors, with a higher dose release and a better conformation with respect to what is achievable with the use of conventional photon or electron beams [fig. 1.1].

![Fig. 1.1: Depth dose curves of different radiations in water [Ref. 1.7].](image)

The very narrow pristine peak (few mm wide) permits to irradiate a well localized region within the body, keeping low the dose release in the proximal (entrance plateau) and distal regions (tail).

To treat an extended target, the longitudinal width of the peak has to be consequently increased by a proper superimposition of several weighted Bragg peaks obtained by modulating the energy of the incident particles [Ref. 1.8]. The resulting depth dose distribution, the so called SOBP (Spread Out Bragg Peak) [fig. 1.2], presents a reduced peak to plateau dose ratio with respect to the
single mono-energetic case, but is still better than the one achievable with conventional radiotherapy.

![Graph showing dose vs penetration depth for a SOBP profile.]

**Fig. 1.2:** A pictorial view of a 1-dim physical Carbon ions SOBP (blue line) as the sum of several pencil beams of different energies and intensities (red lines) [Ref. 1.9].

### 1.2.1 Lateral scattering and Range straggling

For the clinical applications, the lateral scattering of the beam is more important than the longitudinal one because it affects the transverse dose gradient (*penumbra width*), so determining how close to critical structures the beam can get [Ref. 1.10]. Numerous small angle deflections, mainly due to elastic Coulomb scattering of the projectiles with the nuclei of the traversed material, lead to lateral spreading of the incident ions away from the central trajectory. The angular distribution of the scattered particles is roughly Gaussian for small deflection angles and the practical consequence is that the beam spot size is a function of the penetration depth. Furthermore, as shown in figure 1.3, at energies corresponding to common treatment penetration depths, the lateral deflection is more pronounced for lighter ions because multiple scattering effect vary inversely to the particle momentum or on the other hand to the ion mass.
Fig. 1.3: Lateral scattering (FWHM - Full Width at Half Maximum) as a function of the penetration depth in water for different charged particle pencil beams of comparable range and equal initial dimension (FWHM = 4 mm) [Ref. 1.11].

Beams of ions heavier than protons show a better confined lateral broadening (≤1-2 mm): at a penetration depth of 20 cm in water, a ¹²C ion beam having an initial FWHM of 4 mm will only be increased in size to 5 mm whereas a comparable proton beam will double its FWHM [Ref. 1.11]. The measured longitudinal width of the Bragg peak is larger than the one obtained with the mean range formula reported in par.1.2, which is based on the average energy loss of a single particle, due to statistical fluctuations in the energy loss process which cause a dispersion of the path length values around the mean R.

This effect, known as range straggling, increases with penetration depth thus producing, for the same ion type, Bragg peaks of larger width for higher initial energies. In the comparison between different ion species, it can be observed that heavier particles show a narrower Bragg peak with a steeper distal fall off (see figure 1.1), due to the fact that range straggling approximately varies as the inverse of the square root of the particle mass [Ref. 1.12].
1.2.2 Beam fragmentation

When a heavy charged particle beam penetrates through human tissue, it can interact via strong nuclear force giving rise to a fragmentation process that exponentially decrease the fluence of the primary particles as a function of depth, with a corresponding increase of secondary nucleons and fragments. After the nuclear collisions, the remnants of the projectile particles travel nearly in forward direction with almost the same velocity of the incident ones and may therefore cause further fragmentation reactions. These secondary fragments, due to their lower charge, lose less energy than the primary ones and have a longer range thus originating a significant and undesirable dose release beyond the tumor region known as distal tail (see figure 1.1).

Another effect of the fragmentation can deteriorate the therapy effectiveness: because of the nuclear reaction kinematics, the fragments emission angle is larger than the beam lateral scattering and this worsen the physical selectivity in the transverse dimension, especially at the distal side of the Bragg peak where the primary projectiles are stopped and the dose deposition is due to fragments only (see figure 2.2).

It must be stressed that the amount of produced fragments increase with increasing mass and charge of the projectile beam particles and this imposes a limit to the use of ions heavier than Neon in particle therapy [Ref. 1.10].

1.3 Beam delivery systems

In the particle therapy facilities realized so far, different methods are used to deliver the dose distribution as much as possible conform to the tumor shape, spreading the beam both in lateral and distal directions [Ref. 1.11].
For this purpose two strategies have been applied: passive beam shaping was the first to be developed and is still the most commonly used while active scanning is nowadays less diffused, but is becoming the preferred choice for future planned or under construction ion beam therapy facilities.

Both methods have their advantages and disadvantages: passive delivery techniques are safer and simpler, but are not recommended for ion therapy because of the fragmentation induced by the material positioned in the beam’s path; active systems allow a much better dose conformity to the target volume because they are able to vary the particle fluence as a function of beam positions, but are more resources demanding.

While in this work we focused on the active beam delivery system, due to its larger future diffusion, is worthwhile to describe the main features of both techniques.

### 1.3.1 Passive shaping

In the passive systems, the accelerator produces a beam of fixed energy, generally high enough to penetrate at least 20 - 30 cm in human tissue, which is then longitudinally modulated to create the Spread Out Bragg Peak by mean of a variable degrader such as a rotating wheel with different absorber thickness (spiral range shifter) or a ridge (ripple) filter. These materials, put between the beam source and the target, produce variable energy loss, and consequently modulate the range of the primaries, so to longitudinally translate the dose deposition region [Ref. 1.13].

Finally, the small sized incident beam has to be spread out laterally to cover homogeneously the whole target extension. This can be realized using a double scattering system (an occluding ring between two scatterers) and a collimator.
Moreover a special bolus or compensator, individually tailored to the patient target volume, is also routinely used to correct for tissue density inhomogeneities, especially when critical structures are close to the distal edge of the treated volume.

A typical passive shaping set up is reported in the following figure 1.4 which shows the irradiation system of the Tsukuba (Japan) proton therapy facility.

**Fig. 1.4:** The Tsukuba (Japan) proton therapy facility passive shaping irradiation system [Ref. 1.11].
1.3.2 Active scanning

The use of degraders to shift the range, typical of the above described passive techniques, can be avoided if the accelerator allows a swift variation of the primary beam energy. Cyclotrons are unable to change the energy quickly, while modern synchrotrons have in-built flexibility to adjust the energy from cycle to cycle, i.e. within few seconds [Ref. 1.11].

The active beam shaping system takes advantage of the electric charge of the incident particles and use magnetic deflection to change the beam coordinates on the transverse direction.

Three different dynamic scanning techniques are used: wobbling, raster and spot scanning [Ref. 1.14].

**Wobbling** is a dynamic lateral beam deflection technique, developed by Chu et al. in Berkeley [Ref. 1.15], which uses a pair of dipole magnets to produce sinusoidal fields, phase shifted by 90°. In this way, the beam is moved to produce several rings of different radii and doses and, finally, the centre is painted so that a defined closed area is filled.

Another approach to dynamically produce extended and homogeneous dose distribution is the **raster scanning** [Ref. 1.16, 1.17], which is clinically applied at the German carbon ion facility GSI (Gesellschaft für Schwerionenforschung). It uses dipole magnets to move, within every iso-energy slice of the target volume, the small pencil beam along a 2-dimensional continuous path in order to paint the tumor area. This can be achieved performing the scanning with a higher frequency (≈ 40 Hz) in one direction and a lower frequency (≈ 1 Hz) in the perpendicular one, without interrupting the irradiation while moving the beam position [fig. 1.5].
The last active technique is the so called *spot scanning* or *voxelscan* developed at the Paul Scherrer Institute (PSI) [Ref. 1.18]. The beam is magnetically deflected on a grid of positions in a intermittent mode: it is maintained ON until the prescribed dose is reached and then is switched OFF while moving to the adjacent position, whose distance has to be shorter than the beam spot size in order to obtain the sufficient uniformity. In this way for each point of the volume approximately 5-6 beam position contribute to the same extent so to create an “overlap effect” of the dose distribution [fig. 1.6] thus, varying the dose to each spot appropriately, intensity modulated irradiation can be realized and tailored to any irregular shaped target.

**Fig. 1.5:** Schematic drawing of the 2-D magnetic raster scan system based on the virtual dissection of the treatment volume into constant beam energy slices [Ref. 1.11].

**Fig. 1.6:** Schematic drawing of the contributions of different beams position to the covering [Ref. 1.10].
1.4 The biological basis of ion beam therapy

The goal of cancer radiotherapy, the so called *tumor local control*, is the inactivation of all the tumor cells with the simultaneous sparing of the surrounding normal tissue. This means that maximum dose has to be released to the tumor while the dose to normal tissue should be maintained under the tolerance limits, so to minimize the NTCP – *Normal Tissue Complication Probability*.

It has to be stressed that the irradiation effectiveness is a complex quantity that does not depend only on the average absorbed dose, defined as the average imparted energy per unit mass [Ref. 1.19], but also on biological parameters as the *Oxygen Enhancement Ratio* (OER) and the *Relative Biological Effectiveness* (RBE).

As described later, it is the combination of a high *Linear Energy Transfer* (LET) value and of an increased biological effectiveness that makes heavy ion beams biologically more effective in cell killing than low LET radiation as photons, electrons and protons.

1.4.1 LET, OER and RBE

For charged particles, the unrestricted *Linear Energy Transfer* $L_\infty$ is defined [Ref. 1.19] as the ratio of the energy lost due to electronic collisions, $dE$, to the traversed distance, $dl$, and is usually expressed in keV/µm.

For ion beams, the LET value increases significantly with depth, as the particle energy decreases and at the same time the energy deposition is restricted to small volumes along the particle trajectory. Consequently, at the end of the ion path, the local ionization density is very high and DNA lesions are produced
close together in the form of clusters of *double strand breaks* (dsb) which are very difficult to repair [Ref. 1.10].

The presence of oxygen in cells is another factor that modifies the biologic response to the irradiation. In scarcely vasculated (hypoxic) tumor tissue, the oxygen level is low and this reduces the killing effect of the radiation that is mainly due to induced toxic free radicals (*indirect mechanism*). The *Oxygen Enhancement Ratio* (OER) is defined as the ratio between the doses needed to obtain the same biological effect in hypoxic and well oxygenated cells and, in first approximation, is a decreasing function of LET. Electromagnetic low LET radiation thus have a high OER, between 2.5 and 3.2, while for high LET heavy ions, especially in the Bragg peak region, the OER value is near to 1, due to the fact that biological effects are mainly due to the induced DNA lesions (*direct mechanism*) [Ref. 1.20].

This is another advantage for highly ionizing particles, compared to proton or conventional photon beams, because their effect is almost constant for well oxygenated or hypoxic tumor cells.

Due to the complexity of the biological response to radiation of different kind, energy, charge, etc., in order to correctly compare their clinical effects, another parameter has to be introduced: the *Relative Biological Effectiveness* (RBE).

It is defined as the ratio of the absorbed dose $D_\gamma$ of a reference photon beam (typically 250 kV X rays) to the absorbed dose $D$ delivered by the considered particle radiation which produces the same biological effect [Ref. 1.21].

The RBE is not a constant value and cannot be uniquely defined for a given radiation because it depends not linearly on several parameters: absorbed dose level, dose per fraction, energy and type (atomic number) of the ion, desired clinical effect (*endpoint*) and cell/tissue characteristics such as repair capacity and oxygen supply.
In the case of light ions like carbon and due to the above described track structure, the RBE increases with depth or decreasing particle energy, reaching a maximum at 10-20 MeV/u at the end of the particle range. Therefore, to obtain a homogeneous radiation-quality weighted dose (isoeffective dose \( D_{\text{isoE}} \)) \[\text{Ref. 1.22}\] at the level of the SOBP, the absorbed dose should be modulated with a progressive decrease in depth as shown in the following figure 1.7.

![Fig. 1.7: Comparison of the absorbed dose (D) and isoeffective dose (\( D_{\text{isoE}} \)) variations with depth for a 290 MeV/u carbon ion beam treatment with fractionated doses of a planning target volume (PTV) located between 100 and 160 mm in depth (HIMAC - Chiba, Japan) \[\text{Ref. 1.22}\].](image)

The computation of this modulation is a formidable task and is one of the main items that an ion Treatment Planning System must face.

\[1\] The isoeffective dose is given by the product between the absorbed dose and the isoeffective dose weighting factor: \( D_{\text{isoE}} = D \times W_{\text{isoE}} \); see paragraph 2.1.2 for further details.
1.4.2 Radiobiological rationale and patient selection for ion beam therapy of cancer

Considering his biological effect, the use of high LET ion beams in cancer therapy has several advantages: a reduction in the oxygen enhancement ratio (OER), an increase in biological effectiveness (RBE), a reduction of the variation in radiosensitivity related to the position of the cells in the mitotic cycle and a lower effect of dose fractionation. The latter may permit ion treatments with a reduced number of fractions, thus leading to increased efficiency of machine utilization and patient convenience [Ref. 1.23].

These features are particularly important because may aid in the selection of tumor types which are likely to selectively benefit from treatment with light ion beams.

Reduced value of OER and increased RBE may be useful in the treatment of tumors that possess radiation resistant regions containing hypoxic cells, while the reduction in the variation of cell sensitivity may also be advantageous in the treatment of slowly growing tumors. Furthermore, because high LET radiations permit less cellular repair than low LET ones, they can be expected to be selectively more efficient against tumor cells with a high repair capacity, for example, prostate cancer.

Most of the clinical data available today for high LET radiations were obtained in the 1960s with fast neutrons, the first hadrons used for radiation therapy, but are still valid and consistent with more recent radiobiological and clinical findings.

These results, together with the ones achieved in the actually working hadrontherapy facilities, provide useful information to select tumor types and/or body sites which will benefit from ion beam therapy such as, for example:

- salivary gland tumor,
- adenoid cystic carcinoma,
prostatic adenocarcinomas,
slowly growing bone and soft tissue sarcomas,
skull base chordomas and other brain lesions,
early stage hepatocellular carcinoma,
rectum and womb tumors,
some early stage and locally advanced non small cell lung cancer (NSCLC) [Ref. 1.22].

1.5 The world hadrontherapy facilities and the Italian CNAO project

From the first proposal of Robert Wilson in 1946 to the actual successful cancer treatments a long way of research and development has been covered. Pioneering work was first performed with particle accelerators that had originally been built for nuclear physics experiments, such as the BEVALAC synchrotron in Berkeley or the Harvard cyclotron, and only in 1990 the first dedicated medical accelerator went in operation at the Loma Linda University Medical Centre (LLUMC).

Since then many hospital based treatment centers have been built around the world, especially in the United States, in Japan and in Europe, and more than 63’000 patients (at the end of March 2008) have been treated with charged hadrons [Ref. 1.24] as shown in detailed in Appendix A1.

Most of the obtained clinical data are related to proton therapy (more than 55’000) while the experience with carbon ions is much more limited (4’450) due to the fact that, right now, only 3 facilities are in operation: HIMAC-NIRS at Chiba and HIBMC at Hyogo in Japan, and GSI at Darmstadt in Germany.
Carbon ion therapy centers are less diffused because of heaviest hardware conditions due to the highest magnetic rigidity and greater energies requested for the accelerator and the need of bigger magnets for the beam lines. These facilities are consequently more expensive and more difficult to build and operate than the proton ones. Nevertheless, in the last decade, the implementation of carbon ion therapy has demonstrated to be of great interest and several new facilities are under construction or in planning stages in Europe: HIT in Heidelberg (Germany), CNAO in Pavia (Italy), Med-AUSTRON in Wiener Neustadt (Austria), ETOILE in Lyon (France) and PTC in Marburg (Germany) [Ref. 1.25].

The CNAO, National Centre for Oncological Hadrontherapy, will be the first hadrontherapy hospital in Italy that will deliver treatments with active scanning; his construction is almost finished and the commissioning of the LEBT (Low Energy Beam Transfer) line has started with success [Ref. 1.26]. It will be equipped with a 7 MeV/u injector linac and a 25 m diameter synchrotron capable of accelerating carbon ions up to 400 MeV/u and protons up to 250 MeV. Four treatment lines (3 horizontal and one vertical), in three treatment rooms, and a dedicated facility for clinical and radiobiological researches are foreseen in the first stage, but the facility project include also, as a future upgrade, space for an expansion to host two additional rooms eventually equipped with gantries.

The overall number of treated patients will obviously depend on the adopted fractionation scheme but the actual dimensioning of spaces for patients, personnel and people are adequate for about 3’000 patients per year [Ref. 1.26], thus fulfilling the need for carbon ion treatments of radio-resistant tumors defined by epidemiological studies based on Italian tumour registries [Ref. 1.27].
CHAPTER 2

TREATMENT PLANNING SYSTEM FOR CARBON ION THERAPY

2.1 General description of a Treatment Planning System (TPS) for ion therapy

The Treatment Planning System (TPS) is a complex software that has in input the patient diagnostic images and the oncologist prescriptions, and gives as output the direction, kinetic energy and fluence for each elementary beam of the radiation field, in order to achieve the prescribed dose distribution in the planning target volume (PTV) while maintaining the dose to surrounding healthy tissues and organs at risk (OAR) under the tolerance limits.

As first step the system acquires the patient diagnostic images, usually a set of CT (Computed Tomography) or MR (Magnetic Resonance) slices, in DICOM format and, after the radiation oncologist has contoured the volumes of interest (VOI) [Ref. 2.1, 2.2], it builds up the 3-dimensional mathematical model of all these volumes.

As is usually done in conventional photon therapy, the planning target volume is usually generated adding a safety margin of 1-3 mm around the clinical target volume (CTV), to account for any organ movement and uncertainty in the set-up throughout the treatment course.

Finally, given the oncologist prescription dose and its fractionation scheme, the TPS produces the set of directions, fluences and energies of all the pencil beams that optimizes the dose distribution. This distribution can be then checked by the analysis of the output isodose curves and dose volume histograms (DVH), to
verify the consistency between prescribed and released dose and to control the
respect of dose-volume constraints in the critical structures.
The last, more technical and site dependent, part of the TPS transforms the
obtained optimal pencil beams set up in low level instructions to the hardware:
this part will not be treated in the following.

2.1.1 The patient modelling

The anatomical target modelling and structure definition for carbon ion therapy
are very similar to conventional or proton therapy because they are based on the
2-dimensional CT data [Ref. 2.3], usually expressed in Hounsfield units (HU)\(^1\).
On the contrary, an important difference exists in the conversion process to
translate the CT information, representing the X-ray attenuation, into a water
equivalent path length (WEPL).
The CT calibration method which converts HU in electron density to
characterize the radiation energy loss, usually used in conventional
radiotherapy, is insufficient in the case of carbon ion beams [Ref. 2.4]. Light
ions such as \(^{12}\)C, indeed, undergo not only to ionizations and multiple Coulomb
scattering, but also to nuclear reactions so, to convert Hounsfield units in
relative stopping powers and then to accurately define water equivalent tissue
properties, experimental measurements of carbon ranges either on phantom
tissue equivalent materials or real tissues are required [Ref. 2.5]. Beyond that, it
has to be stressed that the simple water equivalent representation is not able to
correctly take into account fragmentation effects that are driven by nuclear
properties of the target, loosely related to the CT electronic density info.

\(^1\) HU or CT numbers are the standard method to assign numerical values to the grey scale of CT
images. The HU scale is a linear transformation of the measured values of the linear attenuation
coefficient of a material \(\mu\) which uses, as reference, the water at STP:
\[ HU = \frac{(\mu-\mu_{\text{water}})}{(\mu_{\text{water}}-\mu_{\text{air}})} \times 1000. \]
2.1.2 Dose calculation algorithms

In hadrontherapy treatments with 3-dimensional active beam shaping, the dose conformation is achieved by the use of many thousands of individually weighted, narrow pencil beams of different energies. Before the dose calculation starts, the target volume is divided into slices, in such a way that each one corresponds to the range of ions at a certain accelerator energy and, doing so, the volumes of interest are represented by a huge number of small volume elements (voxels).

Due to the very large number of degrees of freedom involved in the dose calculation, the use of computer aided “inverse planning” techniques [Ref. 2.6] becomes mandatory to derive, starting from the prescribed dose distribution, the set of optimized beam parameters which produce it. In other words, each voxel gets the dose contribution of several pencil beams with different energies and positions (see paragraph 4.1) and consequently the inverse planning has to optimize the fluences of these beams to adjust the resulting dose to the desired one. This procedure is iterated, for each voxel, by the use of an objective function and subsequent least squares optimization algorithms that yield the final distribution of particle numbers \( N(\text{E}_{\text{beam}}, x, y) \) for each energy slice and scan position [Ref. 2.7].

It must be stressed that, in the case of carbon ion therapy, the optimization has to be done on the biological isoeffective dose \( D_{\text{IsoE}} \), which is the product between the physical absorbed dose \( D \) and a weighting factor \( W_{\text{IsoE}} \) which includes all factors that influence the biological effect and that is approximately given by the best estimate of the local RBE value, for a given dose and specified clinical effects [Ref. 2.8].
2.1.3 Radiobiological modelling for calculation of the isoeffective dose

As already underlined, it is mandatory for ion radiotherapy to incorporate the relative biological effectiveness into routine patient treatment planning [Ref. 2.9]. Since the RBE different dependencies can not be easily parameterized, they have to be experimentally determined or modelled and, at the time of writing, two different strategies have been followed.

At the Japanese carbon ion therapy facilities (HIMAC and HIBMC) the biological treatment planning is adapted to their passive shaping devices and the RBE determination is based on biological in vitro experiments where survival levels were measured and fitted by mean of a linear-quadratic (LQ) model2.

Starting from mono-energetic beams of various incident energies, the dose-averaged coefficients \( \alpha \) and \( \beta \) for a mixed radiation field are then calculated [Ref. 2.10]. Finally, the clinical RBE value = 3.0 in the middle of SOBP of the carbon beam is determined by a link to the enormous National Institute of Radiological Sciences (NIRS) clinical experience with fast neutrons, due to the great similarity of their biological response to the carbon ions one revealed by the performed experiments.

At the GSI facility, on the contrary, M. Scholz and co-workers have developed a computational model (the biophysical Local Effect Model - LEM) [Ref. 2.11, 2.12] that allows to predict the response of biological systems irradiated with particles of a particular type and energy from the knowledge of their survival level to the X-ray irradiation.

The model is mostly based on the assumption that the biological damage in a small sub volume of the cell nucleus is completely determined by the local energy deposition, regardless of the radiation type leading to that energy

---

2 In the LQ model the fraction of surviving cells, S, after a single fraction of dose D is given by the expression: \( S = \exp(- (\alpha D + \beta D^2)) \).
deposition: that is, the difference in biological efficiency between photons and ions arises only from the specific pattern of local dose deposition, i.e. the track structure.

Consequently, provided a detailed knowledge of the number of produced fragments, as well as their energy spectrum, together with the X-ray sensitivity data (α/β ratio) and the radius of the irradiated cells nucleus, the needed values of RBE can be calculated with high accuracy not only within the target region, but all across the irradiation volume.

2.2 The I.N.F.N. – TPS project [Ref. 2.13]

Hadrontherapy has been of explicit interest for the Italian National Institute of Nuclear Physics (INFN) since more than fifteen years, with a significant activity in accelerator technology. Some examples of this interest are the creation of the CATANA proton therapy facility specialized in ocular tumor treatment, and the participation in the construction of the synchrotron for protons and 12C ions at CNAO, the new Italian center in Pavia.

Furthermore, interdisciplinary activities concerning a wide range of related applications have been carried on, such as Boron Neutron Capture Therapy (BNCT), specific radiobiological programs, R&D on mammography and the development of imaging techniques.

Taking into account these various expertises, a further innovative contribute in the field of hadrontherapy could concern the design of a new and advanced Treatment Planning System (TPS). The impact that such an effort can have on radiotherapy is well indicated by the fact that only two TPS are now commercially available: KonRad by Siemens, based on the TRiP98 model developed at GSI for ion treatment with active scanning system, and FOCUS/M commercialized by Mitsubishi Electric Co. for passive system. Both these
products show wide margins for further developments especially for: the modelling of biological effectiveness, the multi field optimization, the implementation of respiratory gating for a future planned 4-dim TPS.

The INFN - TPS project has been divided in five main working areas: nuclear physics for a detailed description of the fragmentation process; experimental radiobiology for the characterization of therapeutic beams; development of new optimization algorithms; realization of dose delivering monitoring tools such as the technique of in-beam PET (Positron Emission Tomography) and, last but not least, the development of Monte Carlo models and tools to obtain a reliable and accurate validation instrument to verify the implemented TPS.

It is just in the last field that the main topics of this thesis will be developed.

2.3 Use of a Monte Carlo code as a TPS validation tool

The use of Monte Carlo (MC) statistical methods for dose calculation represents the most effective tool to verify, and eventually correct, analytical Treatment Planning Systems.

This is due to the fact that, in principle, after an appropriate modelling of the patient body, a MC code can simulate all the dose depositions of all the ions that should be fired by the accelerator, so giving a full verification of the optimization results.

The main items that justify the Monte Carlo choice can be summarized as follows:

- MC can take into account the real composition of human body, going beyond the “water equivalent” approximation;
- MC automatically describes the complexity of mixed radiation fields;
- MC can follow in detail all physics interactions and accurately describe the fragmentation process;
MC can take into account all 3-dim effects such as the backscattering and the angular spread of secondaries;

MC can treat complex geometries including also interfaces between tissues of different density and in the presence of large inhomogeneities such as metallic implants [Ref. 2.14];

in-beam PET monitoring of the irradiation can be performed only using MC simulations to predict the distribution of $\beta^+$ emitters [Ref. 2.15].

Unfortunately all the pros of the Monte Carlo methods are overcome by a big cons that prevents, at the moment, their routine use in clinical practice: a too long computing time is, indeed, required if the MC simulation has to be performed for all the particles required by the TPS optimization results. This problem will be analyzed more in detail later on in this work, together with the explanation of its possible solution.

### 2.3.1 The accurate target modelling

As described in the previous 2.1.1 paragraph, CT images are the input data for the patient body modelling.

Commercial analytical treatment planning systems use a Water Equivalent Path Length (WEPL) approximation to account for density and chemical composition variations between different biological tissues.

The water equivalent approximation transforms the trajectory of an ion from the CT system into a water equivalent system in the beam’s eye view [fig. 2.1], in such a way that higher is the density of the true tissue (bigger squares in the left side of the following picture), longer is the water equivalent distance traveled [Ref. 2.7]. In this way, however, the originally regular shaped target volume will be distorted in the water equivalent system (right side of fig. 2.1).
Moreover, this approach is good for tissues with density close to the water one such as brain, while is less correct when applied to materials that present strong difference from water such as bone or metallic implants.

On the contrary, Monte Carlo codes could be able to take into account the realistic patient anatomy. Before starting the simulation, in fact, a customized program can be used to take the further step that is needed to transform the voxel electron density given by the CT scan in a voxel averaged atomic number, mass number and density.

Of course, also this procedure introduces a certain level of approximation but, nonetheless, represents a good improvement compared to the pure water equivalent approach.

A possible method is to start from the correlation between HU intervals and tissue parameters established by W. Schneider and co-workers [Ref. 2.16], to import the 2-dim CT pixel data and to relate them with 3-dim voxels of the corresponding material in the simulation geometry. In such a way, the mass density and chemical composition of different human tissues/organs can be described and the target volume shape is not distorted.
2.3.2 The detailed fragmentation description

Another item where the use of Monte Carlo codes can be of great help is in taking into account the ion fragmentation process. Commercial TPS for carbon ion beams such as TRiP98, include an analytical model to calculate the energy loss distribution of primary and secondary particles as a function of depth in water or water-equivalent material. The one dimensional depth-dose profiles are calculated in steps of 10 MeV/n, neglecting the multiple Coulomb scattering of the beam and of secondary fragments and all particles generated during the $^{12}$C slowing down such as $\gamma$-rays, and $\delta$-rays [Ref. 2.7].

Using Monte Carlo simulations, instead, a detailed 3-dimensional description of all the complex processes of the ion interaction with matter can be achieved. This includes all nuclear reactions, from pure fragmentation process at highest energies to multiple Coulomb scattering, deep inelastic scattering and fusion at the lowest ones.

In such a way, the production and transport of all secondary lower charge fragments (i.e. H, He, Li, Be, B), which broaden the irradiation field and mainly contribute to the distal dose deposition, can be accurately take into account.

In the following figure 2.2 an example of MC simulated $^{12}$C depth-dose profile (top), with a detailed view of the different secondary fragment contribution to the distal tail (bottom), is reported together with experimental data [Ref. 2.17] showing a very good agreement between the two data sets.

The presented simulation results have been obtained by A. Mairani in his Ph.D. thesis [Ref. 1.9] using a development version of the official release of the FLUKA Monte Carlo code [Ref. 2.18, 2.19], while the experimental data are reported by courtesy of E. Haettner, D. Schardt et al. from the GSI facility and K. Parodi and S. Brons (HIT).
Fig. 2.2: Depth dose profile of a 400 MeV/n carbon beam on water (top). The solid lines represent the simulated contribution from primary $^{12}$C ions and secondary fragments (detailed view in the bottom), while the black points give the experimental data [Ref. 1.9].

2.3.3 The statistical accuracy in dose calculation

The outcome of analytical treatment planning systems, which is a numerical optimization of the absorbed dose, is generally given without a statistical error. The 3D statistical dose displayed information are: 3D dose maximum value and maximum, mean and minimum dose in the planning target volume.

With Monte Carlo simulations an optimal statistical accuracy (i.e. few % or less) can instead be achieved but, as previously mentioned, the long computing
time (i.e. in the order of 0.1 sec mean CPU time for primary ion\(^3\)) prevents, at the moment, their direct use in clinical routine to calculate a treatment plan. If the MC simulation has to be performed for all the particles that form the TPS optimized treatment field, i.e. order of hundred millions, then the resulting computing time will be in the order of about ten days. The CPU total requested time is then far away from the one, in order of about ten minutes, which is generally necessary, in clinical practice, to an analytical TPS to produce his optimized treatment plan.

Nevertheless, a possible solution to this problem can be found in order to use the Monte Carlo code as a reliable tool for the validation and the improvement of analytical treatment planning systems.

A statistically significant dose calculation can, indeed, be obtained simulating only a sub-sample, for instance 1/100, of the total number of particles (\(10^8 - 10^{10}\)) used by the TPS to irradiate the target volume and to achieve the dose prescription. The MC simulation can be stopped when the calculated fractional (or relative) uncertainty, that is the relative fluctuation \(d\bar{E}/\bar{E}\) of the average deposited energy (or dose), is less or equal to 2-3 %.

This condition must be particularly satisfied in each voxel of the irradiated geometry belonging to interface regions between different tissues or near to a critical structure such as an organ at risk.

Moreover, it has to be stressed that the simulation accuracy is only one of the uncertainties that have to be considered in the overall treatment plan, such as the beam modelling or the patient positioning, and that the resulting global accuracy given by the sum of all the contributions is in the order of 5-15 %.

\(^3\) The reported CPU time/primary ion was measured on a single core state of the art computer (Pentium IV running at 3.4 GHz).
2.3.4 Monte Carlo optimization tools development

In the previous paragraph we showed that the necessary condition for the use of Monte Carlo codes in TPS validation is to reach the needed statistical accuracy in a fairly reduced CPU time. It’s then self evident that methods of improving the simulation efficiency are of particular interest for the topic of this work.

The efficiency of a Monte Carlo calculation is defined as \( \varepsilon = \frac{1}{\sigma^2 T} \), where \( \sigma^2 \) is an estimate of the variance on the quantity of interest (i.e. the dose) and \( T \) is the CPU time required to obtain this variance [Ref. 2.20]. The computation time \( T \) is obviously proportional to the number \( N \) of statistically independent simulated particles, while \( \sigma^2 \) decreases as \( 1/N \) according to the central limit theorem\(^4\).

The methods of improving the statistical efficiency of MC simulations can be classified in two different categories:

a) *Variance Reduction Techniques* (VRT) that reduce \( \sigma^2 \) for a given \( T \) (or \( N \)) without altering in any way the physics, thus producing unbiased results;

b) *Efficiency Enhancing Techniques* (EET) that decrease \( T \) for a given \( N \) while not changing the variance, by making an approximation which may or may not affect the final result in a significant way [Ref. 2.21].

As already stated, the aim of this thesis is to develop new tools to optimize the efficiency of Monte Carlo simulations, in the framework of the INFN - TPS project, for the validation of a treatment planning system for carbon ion radiotherapy with an active scanning technique. As reported in the following chapters, methods have been developed to calculate and minimize both the

---

\(^4\) The central limit theorem (CLT) states that given a random variable distribution with mean \( \mu \) and variance \( \sigma^2 \), no matter his shape, the sampling distribution of the mean approaches a normal distribution with a mean \( \mu \) and a variance \( \sigma^2/N \) as \( N \), the sample size, increases.
overall time of processing and the number of histories (i.e. of primary ions) that makes up every pencil beam of the treatment field.

2.4 The choice of the FLUKA Monte Carlo code

In the framework of the INFN - TPS project the FLUKA Monte Carlo code has been chosen as a reliable instrument for the validation of analytical treatment planning systems because it has been already successfully applied to proton therapy [Ref. 2.22] and has shown to be able to meet the following requirements:

- capability of being coupled to CT scans to import the irradiation geometry and to accurately identify the target organs in terms of density and chemical composition;
- capability to embed the radiobiological model used by the TPS;
- availability of reliable nuclear models for transport and interactions of all components of the mixed radiation field (ions, hadrons and electromagnetic particles), including nucleus-nucleus reactions.

2.4.1 FLUKA code capabilities

FLUKA is a multi purpose MC code for calculation of particle transport (60 different types plus heavy ions) and interactions with matter that was originally designed for high-energy physics, and later extended to cover a wider range of energies and applications including radiation therapy, accelerator shielding, dosimetry, activation, neutrino physics and space radiation [Ref. 2.23].

Arbitrarily complex geometries can be described in the code using combinatorial functions applied to elementary objects (bodies) such as solids
and infinite planes; voxel geometries are supported as well, thus allowing to properly model CT scans for treatment planning purposes.

The algorithm used for voxel geometries is optimized in order to minimize memory requirements (2 bytes per voxel only) and employs dynamically allocated structures to speed up the tracking process.

Different physical models are implemented in order to accurately describe all kinds of processes such as hadron-hadron, hadron-nucleus, nucleus-nucleus, electromagnetic and \( \mu \) interactions.

Recent developments include also the implementation of the BME (*Boltzmann Master Equation*) event generator that allows the description of light ion interactions on material of biological interest, in the energy interval from the Coulomb barrier up to 100 MeV/\( n \). This extends the FLUKA energy range of heavy ion interactions to a region of utmost importance for hadrontherapy, in order to properly describe the fragmentation process in the Bragg peak region.

Moreover, as reported in the work by F. Sommerer et al. [Ref. 2.24], the precise determination of ion range and ionization losses can be performed thanks to an original multiple Coulomb scattering algorithm [Ref. 2.25] and to a statistical approach alternative to the standard Landau and Vavilov ones that provides a very good reproduction of average ionization and of fluctuations [Ref. 2.26].
CHAPTER 3

DELTA RAYS CONTRIBUTION ANALYSIS

3.1 Delta rays contribution

Aim of the following reported work is to analyze and quantify the contribution of secondary electrons to radial dose distribution and Monte Carlo simulation computing time in the case of $^{12}$C ion beam interactions in a water phantom.

According to the $\delta$-ray theory of track structure, the secondary electrons ejected from the medium by the passing ion with sufficient energy to ionize further atoms traveling along their path, i.e. the so called $\delta$-rays, are the main responsible for the radiation damage induced in the traversed material [Ref. 3.1].

For an incident particle with mass $M$ and momentum $M\beta\gamma c$, the maximum kinetic energy $T_{\text{MAX}}$ which can be imparted to a free electron with mass $m_e$ in a single collision, is given by the following expression [Ref. 3.2]:

$$T_{\text{MAX}} = \frac{2m_e c^2 \beta^2 \gamma^2}{1 + 2m_e \gamma / M + (m_e / M)^2}$$  \hspace{1cm} (3.1)

When $2m_e \gamma / M << 1$, such in the case of Carbon ions, the above relation can be further simplified and the following “low energy” approximation is used:

$$T_{\text{MAX}} = 2m_e c^2 \beta^2 \gamma^2$$  \hspace{1cm} (3.2)

Using this relation the $\delta$-rays maximum kinetic energy has then been determined for an incident $^{12}$C ion beam with kinetic energy varying between 80 MeV/n and 430 MeV/n corresponding to a range in tissue between 20 mm and
330 mm respectively: resulting values are included in the interval ranging from 180 keV to 1.16 MeV.

It has to be stressed that the production of high energy secondary electrons is rather low probable and that their energy release is quite low compared to the carbon ions one, due to the \( z^2 \) dependence of the electronic stopping power formula (see paragraph 1.2). Nevertheless the \( \delta \)-rays contribution to dose deposition is very important due to the huge number of secondary electrons produced by the interactions of the primary ion beam.

Moreover the number of \( \delta \)-rays have a great influence on the Monte Carlo simulation computing time due to the linear proportionality between tracking time and number of produced particle to be followed.

In order to estimate the number of \( \delta \)-ray electrons ejected by an incident carbon ion per unit length of ion path the following formula [Ref. 3.2], representing the distribution of secondary electrons with kinetic energies \( T >> I \) (\( I \) = mean excitation energy = 75 eV in water), can be used:

\[
\frac{d^2N}{dTdx} = \frac{1}{2} K z^2 \frac{Z}{A} \frac{1}{\beta^2} F(T) \quad T^2
\]

(3.3).

Here \( \beta \) and \( z \) are the velocity and the charge of the primary particle while the spin dependent factor \( F(T) \) is given by:

\[
F(T) = 1 - \beta^2 \frac{T}{T_{\text{MAX}}}
\]

(3.4).

Using the above reported expressions 3.3 and 3.4, calculation have then been performed for a carbon ion beam with initial kinetic energy equal to 430 MeV/n, that is the maximum kinetic energy/nucleon actually used in carbon ion therapy facilities.

The resulting secondary electron spectrum, reported in the following figure 3.1, allows to estimate the number of \( \delta \)-rays produced in water by the incident carbon ion as a function of the \( \delta \)-rays kinetic energy.
Fig. 3.1: Secondary electron spectrum from incident $^{12}$C ion beam with initial kinetic energy $E = 430$ MeV/n in water.

### 3.2 Dose deposition profile lateral broadening

As described in the theory of G. Molière [Ref. 3.3], the lateral broadening of an ion beam mainly arises from several and highly probable deflections by small angles as well as rare large angle single scattering events of the projectiles particles with the target nuclei.

Also the kinematics of nuclear reactions contributes to the lateral width of the beam, predominantly at the distal side of the Bragg peak, where the primary ions are stopped and the dose distribution is made up only of the fragments contribution [Ref. 1.10].
The lateral beam broadening also arises from the secondary electrons which have sufficient energy to induce further ionizations in their path. The global effect is clearly evident in the following figure 3.2 which reports the radial total energy distribution produced, with a fine cylindrical mesh, in a water equivalent phantom by a point like carbon ion beam with initial kinetic energy \( E = 250 \text{ MeV/n} \) propagating along the \( Z \) axis.

![Radial total energy distribution](image)

**Fig. 3.2:** Radial total energy distribution in water by a \(^{12}\text{C}\) ion beam with initial kinetic energy \( E = 250 \text{ MeV/n} \).

As expected, the lateral broadening is more important in the distal zone beyond the Bragg peak, which is located at \( Z \) depth \( \sim 12.3 \text{ cm} \), where only fragments survive, than in the region of the \(^{12}\text{C}\) beam dose deposition, represented in red in the picture.

The reported simulation data have been obtained, using the standard FLUKA settings for ion beam runs: tracking down all the particles to 100 keV and
recording their energy, as local dose deposition, when they are no more followed as they go below transport threshold.

Since the accurate tracking of all the particles of the beam down to very low energy is very CPU consuming, the first optimization step is to tune the process, namely the energy cutoff, for all the low energy tracks, especially the $\delta$-ray electron ones. In the following we describe the related FLUKA Monte Carlo optimization.

### 3.3 The FLUKA simulation physics settings

The Monte Carlo simulations of a carbon ion beam interaction with a water equivalent phantom has been carried out using, for optimal performances, the suite of physics settings recommended by the FLUKA code authors for applications to hadrontherapy, which is the default ‘HADROTHERapy’ [Ref. 2.18].

This includes detailed transport of primary $^{12}\text{C}$ ions and secondary particles (e.g., protons, electrons, alphas, neutrons), using the most accurate algorithm for multiple Coulomb scattering of charged particles with the activation of inelastic form factor corrections.

In addition to these defaults the novel, and not yet implemented in the official release of the code, low energy BME nucleus-nucleus generator was activated for an accurate description of the nuclear interactions involved in the fragmentation process.

Different energy thresholds from the ‘HADROTHERapy’ default for electron production and transport have been used in order to investigate their effect on the beam broadening and on the computing time.
Two different cards, EMFCUT and DELTARAY, can be used in the input file of the FLUKA simulation to define kinetic energy thresholds for the production and transport of primary and secondary electrons ($\delta$-rays).

The option DELTARAY concerns only the delta rays production by charged hadrons and muons, while the EMFCUT card can be use to set the energy thresholds for electron, positron and photon production and transport.

Obviously, delta rays produced by electrons and positrons are always generated, provided their energy is larger than the electron production threshold defined by the option EMFCUT.

The analysis of the effect on the beam broadening and on the CPU time has been done using the same kinetic energy threshold for both the above described input cards.

It has to be stressed that the production and transport thresholds don’t affect the accuracy of the dose calculation: if the energy of the produced delta rays is lower than the transport cut off, then the energy is deposited on the spot.

In this way, regardless of the threshold, the energy is conserved along the track.

On the other hand there is an effect on the beam broadening, i.e. the lateral spread and divergence mainly resulting from multiple Coulomb scattering, which is of utmost importance in hadrontherapy because it determines how close to a critical structure the beam can get.

A very rough approximation of this quantity can be achieved considering, as it follows, the electron average path length traveled as it slows down to rest (range) [fig. 3.3], calculated in the continuous slowing down approximation (CSDA) by integrating the reciprocal of the total stopping power with respect to energy.
Using as reference material the ICRU four component soft tissue [Ref. 3.5]
(fraction by weight: H - 10.1 %, C - 14.9 %, N - 2.6%, O - 76.2 %), which
presents the same density and similar average ionization potential of the water
target material used in the simulations and, as guideline, the related values of
the electron range, seven different energies have been chosen in the interval
from 100 keV (that is the ‘HADROTHErapy’ default) up to 10 MeV, as shown
in the following table.

<table>
<thead>
<tr>
<th>Energy threshold</th>
<th>Electron range</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 keV</td>
<td>0.14 mm</td>
</tr>
<tr>
<td>200 keV</td>
<td>0.45 mm</td>
</tr>
<tr>
<td>350 keV</td>
<td>1.07 mm</td>
</tr>
<tr>
<td>500 keV</td>
<td>1.78 mm</td>
</tr>
<tr>
<td>800 keV</td>
<td>3.33 mm</td>
</tr>
<tr>
<td>1 MeV</td>
<td>4.41 mm</td>
</tr>
<tr>
<td>10 MeV</td>
<td>5.02 cm</td>
</tr>
</tbody>
</table>

Tab. 3.1: Chosen simulation energy thresholds and corresponding electron CSDA
range in the soft tissue.
3.4 The delta rays threshold effect on beam broadening

In the FLUKA Monte Carlo code, the ‘USRBIN’ input card can be used for scoring energy deposition in a regular spatial structure (binning), which is geometry independent i.e., not constrained by region boundaries and not affecting the transport step size.

Several simulation runs have been performed using two different meshes: Cylindrical with bins extension $\Delta R = 200 \, \mu m$ and $\Delta Z = 5 \, mm$, and Cartesian with $\Delta X = \Delta Y = 300 \, \mu m$ and $\Delta Z = 5 \, mm$.

A customized FLUKA executable has been generated to separate the different particles i.e., carbon ions, alpha, protons and electrons contributions to the total energy deposition.

Moreover, to analyze the effect of the energy thresholds on the beam broadening, a custom code based on the interactive graphical data analysis library PAW (Physics Analysis Workstation) [Ref. 3.6] has been developed.

Many simulations have been performed using the threshold energy values reported in table 3.1 and three different initial $^{12}\text{C}$ ions kinetic energies: 150, 250 and 350 MeV/n.

Scoring the deposited energy in the above described Cartesian mesh and using the developed code, the following reported results have been obtained.
Fig. 3.4: Gaussian fit of the total energy distribution in water for the simulation run with initial $^{12}$C ions kinetic energy $E = 250$ MeV/n with delta rays threshold energy equal to 100 keV.

In the previous figure 3.4 the transverse profile of the total energy distribution, for the simulation run with incident beam energy $E = 250$ MeV/n and delta rays threshold energy $= 100$ keV, have been fitted with a Gaussian function.
The reported plots have been taken for depth $Z = 0.5, 7.5, 12.5, 15$ cm, that is in correspondence of the entry point, the entrance plateau, the Bragg peak and the distal tail.

An equal procedure has been followed also for the single particle contribution to the overall deposited energy, for different delta rays thresholds and for the other incident beam energies.

For comparison, similar plots are shown in the following figure 3.5 for delta rays threshold energy equal to 10 MeV and for $Z = 12.5$ and 15 cm.

Fig. 3.5: Gaussian fit of the total energy distribution in water for the simulation run with initial $^{12}$C ions kinetic energy $E = 250$ MeV/n with delta rays threshold energy equal to 10 MeV.

As can be seen, the resulting Gaussian profiles are very similar and the RMS value of the histograms and the resulting fitting parameters present only negligible difference.
In order to obtain a more quantitative analysis of the effect of delta rays threshold on the beam broadening, in the previous figure 3.6 are shown the sigma values (cm) of the lateral total energy distribution as a function of the longitudinal coordinate $Z$ (cm).

As can be seen, the fitting sigma values of the total energy distribution are not sensible to the delta ray thresholds, even for the limit values of the delta rays threshold energies used in the simulations.

Similar results, not reported, have been obtained also for all the other simulated incident beam energies.

The global lack of effect of delta rays thresholds on beam energy deposition broadening can be understood considering that, compared with the primary $^{12}$C ions and other lighter fragments such as protons and alphas, the electrons make only a tiny contribution to the dose deposition.
3.5 CPU time improvement for different delta rays thresholds

In the standard output file of the FLUKA simulation it is reported the information about the ‘average CPU time per history’ i.e., the mean time required to follow a primary particle.

The collection of the CPU data for all the simulations performed with seven different delta rays threshold energies and three incident beam energies, has allowed the evaluation of the effect that the energy thresholds have on the computing time.

In the following figure 3.7, the decreasing values of average CPU time per history for increasing delta rays threshold energies are reported with different color corresponding to each of the simulated incident beam energy.

As expected, a reduced value of the energy threshold corresponds to a higher value of the time needed to properly describe the primary $^{12}$C ions interactions.

![Graph showing average CPU time per primary for different delta ray thresholds](image)

**Fig. 3.7:** Average CPU time per history for all simulation set ups.
The maximum average CPU time per history, \( \sim 0.26 \) sec, is reported for the simulation with the higher incident beam energy, 350 MeV/n, and the lower delta rays threshold energy, 100 keV.

In order to quantify the CPU time improvement achievable with a proper choice of the delta rays production threshold, the ratio between the time values corresponding to the 100 keV and the 10 MeV thresholds have been calculated and plotted as a function of the initial \(^{12}\text{C}\) ions kinetic energy [**fig. 3.8**].

![CPU time ratio](image)

**Fig. 3.8:** Average CPU time ratio for the considered limit values of delta rays production threshold.

The resulting average CPU time ratios increase with the value of the incident beam energy from 4.6 for the 150 MeV/n \(^{12}\text{C}\) ions up to 5.9 for the 350 MeV/n ones.

Given these results, it can be asserted that a significant reduction of the computing time can be achieved raising the delta energy production threshold or
switching off the production process, without compromising the accuracy of the simulation.

Consequently, from now on, the simulation runs will be performed using a delta rays energy threshold equal to 10 MeV.

This choice permits to gain a variable factor from 4.5 to 6, depending on the primary beam particles energy, in the resulting computing time with respect to the FLUKA standard ‘HADROTHErapy’ setup.
CHAPTER 4

TPS PARTICLE FLUENCE VERIFICATION AND OPTIMIZATION

4.1 Mathematical bases

As previously reported (see paragraph 2.1.2), in a carbon ion radiotherapy treatment with a 3-dimensional active scanning shaping system the radiation field is produced by superimposing many elementary weighted pencil beams, both in longitudinal and in transverse direction [fig. 4.1].

![Fig. 4.1: Schematic drawing of the different dose contributions to a given voxel (i, j, k) in a 3D active scanning shaping system.](image)

In this way the total absorbed dose in any voxel of the irradiated geometry is given by the sum of the contributions coming from every pencil beam of the treatment field aiming at that voxel or at his neighbors.
If we consider a 3-dimensional target with NBINX voxels along the $x$-axis, NBINY along $y$ and NBINZ along $z$, then the total absorbed dose (or imparted energy) in a given elementary volume $(i, j, k)$ can be written as

$$D(i, j, k) = \sum_{\alpha} N_{\alpha} \times d_{i,j,k}^{\alpha}$$

where $N_{\alpha}$ is the particle number of the $\alpha$-th pencil beam and $d_{i,j,k}^{\alpha}$ is the dose/primary deposited by the pencil beam $\alpha$ in the voxel $(i, j, k)$.

The pencil beam fluence is given by the ratio between the number of incident ions and the beam cross section. If we assumed that all the pencil beams have the same spot size of radius $\sigma_r$, then there is a simple proportionality relation between the fluence and the total number of ions of $\alpha$-th pencil beam:

$$\varphi_{\alpha} = \frac{N_{\alpha}}{\pi \sigma_r^2}.$$

In this hypothesis the fluence and the particle number of each beamlet play the same role.

In order to achieve the desired dose distribution, the treatment planning system thus solves the so-called “inverse problem” that is, starting from the prescribed dose to the tumor volume previously contoured by the oncologist, it gives the values of the unknown fluences $N_{\alpha}$, together with the other beam parameters, as output of his optimization procedure.

In analytical TPS the dose calculation relies on a set of measured PDD (Percentage Depth Dose) curve in water for different beam energies, and the particle number optimization is done by the iterative use of minimization algorithms (Conjugate Gradient, Quasi Newton, Simulated annealing) until the predefined dose in the target volume is reached, that is until the difference between the calculated dose and the prescribed one is minimal.
It has to be stressed that the TPS calculation of the particle number is made using a water equivalent approximated model of the patient body. This approximation can be not enough accurate especially on the boundaries between materials of very different density and, moreover, can be really crude in cases where the nuclear effects (i.e. the $^{12}$C fragmentation) must be taken into account.

On the contrary, the Monte Carlo code embeds the correct target geometry thanks to a translation of the CT info to different materials [Ref. 2.16], including their atomic composition. This translation gives, of course, a much better approximation of the real patient body.

It is then well understood the value of the MC validation of the TPS results. However, the use of Monte Carlo codes as a reliable tool for the validation of the dose calculation performed by the analytical TPS is useful only if the results of the simulation are given with the needed accuracy level and in an acceptable time for the clinical routine.

Following the suggestions of the AAPM (American Association of Physicists in Medicine) Task Group n. 65 for conventional radiotherapy using MV photons [Ref. 4.1], it can be said that accepting an overall desired accuracy equal to 5% in the dose delivered to a volume, then the accuracy level of computed dose distributions should be between 1% and 2%.

It must be stressed that larger errors should be expected when more complex treatment techniques, such as the hadrontherapy, are applied.

Given these hints it has been decided to fix, in the following reported Monte Carlo simulations, a target value of the dose accuracy level, i.e. a relative uncertainty $R_i^0 = \frac{\sigma_{<E>}}{<E>}$ of the average value of the total deposited energy in each voxel of the irradiated geometry equal to 2%.
If we consider a generic hadrontherapy treatment with a physical dose on the order of some tens Gray in a few cm$^3$ extended target volume, this translates in a total number on the order of $10^8 - 10^{10}$ carbon ions delivered by the beam system and accordingly calculated by the TPS. Due to this very huge number of carbon projectiles, the statistical fluctuation of the actual delivered dose, scaling with $\frac{1}{\sqrt{N}}$, is negligible. On the other hand, the simulation of such an amount of ions would take a time much longer then what is tolerable for the clinical routine (less than 1 hour) even using clusters with many CPUs.

The simplest receipt to achieve the goal of a clinical routinely use of MC simulations for TPS validation is to introduce of a scale factor $f_{\text{global}}$, common for the fluence of all pencil beams, given by the ratio between the number of primary carbon ions $N^{MC}$ needed to be actually simulated by the Monte Carlo code and the total number of particles used to irradiate the target volume, i.e. $N^{TPS}$, according to the output of the analytical TPS.

A first raw approximation of this factor can be obtained considering that, as following demonstrated, the relative uncertainty $R$ is proportional to the inverse square root of the particle number $N$. If $R^0$ is the desired accuracy to be obtained on the average voxel dose and $R_{\text{max}}^{TPS}$ is the maximum accuracy of the voxel doses obtained with the true $N^{TPS}$ particle number (usually much smaller than $R^0$), the global scaling factor is given by the following expression:

$$N^{MC} = N^{TPS} \cdot f_{\text{global}} = N^{TPS} \left( \frac{R_{\text{max}}^{TPS}}{R^0} \right)^2$$

Beyond this simple and global method to obtain a simulation with acceptable statistical accuracy, a more complex procedure that uses different scaling factor for each pencil beam of the treatment field will be presented.
The developed method, following reported in this work, is used to first verify directly the number of $^{12}$C ions, $N$, for every pencil beam, $\alpha$, computed by the TPS, and then to find a scale factor for each pencil beam that allow to achieve, by the simulation, the prescribed dose value in the planning target volume with the desired accuracy level $R^0$.

Due to the fact that the computing time of a MC simulation is linearly proportional to the number of particle histories, the ion fluence optimization procedure obviously involves a reduction of the CPU time.

In the following, the mathematical bases of the $^{12}$C ions fluence calculation and optimization procedure are reported.

For every voxel of the irradiated target, the released energy distribution produced by each pencil beam, $\alpha$, can be described using the first moment $\mu_\alpha$ and the second moment, or variance, $\sigma_\alpha^2$ given by the following expressions:

$$\mu_\alpha = \frac{\sum_{l=1}^{N_\alpha} E_l}{N_\alpha} \quad (4.1)$$

$$\sigma_\alpha^2 = \frac{1}{N_\alpha} \sum_{l=1}^{N_\alpha} E_l^2 - \mu_\alpha^2 \quad (4.2)$$

where $N_\alpha$ is the number of primary $^{12}$C ions of the beamlet $\alpha$ and $E_l$ are the energy releases in the selected bin.

As previously underlined, for each bin (i, j, k) the total absorbed dose is given by the sum of different pencil beam contributions so, in order to calculate the relative uncertainty of the average value of the total deposited energy (or dose), the variance of the total (i.e. due to all pencil beams) released energy distribution has to be determined.
Let’s consider, for instance, a simple case where only two pencil beams with particle number $N_1$ and $N_2$ deposit energy in the given voxel (i, j, k), as represented with different color in the following figure 4.2.

![Figure 4.2: Example of energy release distributions in a given voxel (i, j, k) produced by two different pencil beams.](image)

Then, using the above reported expression 4.1 and 4.2 and taking into account that the total particle number of the irradiation field, $N_{TOT}$, is given by the sum of the pencil beam contribution $N_\alpha$, that is $N_{TOT} = \sum_{\alpha=1}^{N_{pencil}} N_\alpha = N_1 + N_2$, the first moment and variance of the two released energy distributions are given by the following expressions:
\[ \mu_l = \frac{N_1}{N_1+1} \sum_{l=1}^{N_1} \frac{E_i}{N_1} \quad \mu_2 = \frac{N_1+N_2}{N_1} \sum_{l=N_1+1}^{N_1+N_2} \frac{E_i}{N_1} \] (4.3)

\[ \sigma_i^2 = \frac{1}{N_1} \sum_{l=1}^{N_1} E_i^2 - \mu_i^2 \quad \sigma_2^2 = \frac{1}{N_2} \sum_{l=N_1+1}^{N_1+N_2} E_i^2 - \mu_2^2 \] (4.4)

Using these expressions, the following relation can also be obtained:

\[ \sum_{i=1}^{N_1+N_2} E_i^2 = N_1(\sigma_1^2 + \mu_1^2) + N_2(\sigma_2^2 + \mu_2^2) \] (4.5)

Given the above expressions 4.3, the first moment (i.e. the average) of the total energy distribution produced by the two pencil beams in the i-th voxel can then be calculated as follows:

\[ \mu_i = \frac{N_1+N_2}{N_1 + N_2} \sum_{i=1}^{N_1+N_2} \frac{E_i}{N_1} = \frac{N_1 \mu_1 + N_2 \mu_2}{N_1 + N_2} \] (4.6)

On the other hand, using the relations 4.4 and 4.5, also the variance can be obtained:

\[ \sigma_i^2 = \frac{1}{N_1 + N_2} \sum_{i=1}^{N_1+N_2} (E_i - \mu_i)^2 = \sum_{i=1}^{N_1+N_2} \frac{E_i^2}{N_1 + N_2} - \mu_i^2 = \frac{N_1(\sigma_1^2 + \mu_1^2) + N_2(\sigma_2^2 + \mu_2^2)}{N_1 + N_2} - \mu_i^2 \] (4.7)

Thus, considering the general case of \( N_{pencil} \) beams contributions, the variance of the total energy distribution in the i-th voxel can be described as follows:
\[ \sigma_i^2 = \frac{\sum_{\alpha=1}^{N_{\text{pencil}}} N_\alpha (\sigma_{i,\alpha}^2 + \mu_{i,\alpha}^2)}{\sum_{\alpha=1}^{N_{\text{pencil}}} N_\alpha} - \mu_i^2 \]  
\[ (4.8) \]

where \( \mu_i = \frac{\sum_{\alpha=1}^{N_{\text{pencil}}} N_\alpha \mu_{i,\alpha}}{\sum_{\alpha=1}^{N_{\text{pencil}}} N_\alpha} \) is the first moment of the total voxel energy distribution.

In this way the resulting variance \( \sigma_i^2 \) is a function of the particle number \( N_\alpha \) of the different pencil beams which form the treatment field and of the physical parameters \( \sigma_{i,\alpha} \) and \( \mu_{i,\alpha} \) which characterize the energy release distribution, due to each pencil beam \( \alpha \), in the \( i \)-th elementary volume of the irradiated target.

Both the dose and the relative percentage uncertainty \( \frac{\sigma_E}{E} \) in the \( i \)-th voxel can be computed by the above expression 4.8, given the knowledge of the matrix elements \( \sigma_{i,\alpha}^2 \) and \( \mu_{i,\alpha}^2 \). These matrices include all the physics of the energy release process, that is, every element \( \mu_{i,\alpha} \) and \( \sigma_{i,\alpha} \) is given by the convolution of several effects such as energy release, carbon ions propagation, multiple scattering, beam size and energy, etc.

In the reported procedure, \( \mu_{i,\alpha} \) and \( \sigma_{i,\alpha} \) have been computed performing a Monte Carlo simulation of the physical radiation interactions within the target, using a statistically significant sub sample of the total treatment particle number \( N_{\text{TOT}} \). Moreover, the use of specific user written routines (\textit{mgdraw}.f et al.) has been necessary in order to properly calculate the value of the energy release in each voxel of the irradiated geometry.
While the use of the MC to obtain $\mu_{i,\alpha}$ and $\sigma_{i,\alpha}$ has the advantage of correctly taking into account all the physics, the target geometry and the beam setup, on the other hand it gives values of these matrix elements that have embedded statistical fluctuations. The number of events that have to be simulated, i.e. the dimension of the $N_{TOT}$ sub sample, must then be chosen in such a way that $\mu_{i,\alpha}$ and $\sigma_{i,\alpha}$ have a limited statistical fluctuation, as will be shown in the following paragraph.

### 4.2 The Monte Carlo fluctuations of $\mu_{i,\alpha}$ and $\sigma_{i,\alpha}$

In order to check the correlation between statistical fluctuations and number of beam particles which forms every pencil beam to be used in the pre-run calculation of $\mu_{i,\alpha}$ and $\sigma_{i,\alpha}$, several simulations have been performed dividing each of them into 20 batches (or runs). The used simulation geometry set up is a simple water phantom with NBINX = NBINY = 10 and NBINZ = 20 bins with dimension equal to $1 \times 1 \times 1$ cm$^3$, and the initial particle number of each pencil beam has been varied so that $N_{\alpha} = 1'000$, 10'000 or 50'000.

The resulting values of the average deposited energy $\langle E \rangle = \mu$ and of the standard deviation $\sigma_E$ have been collected for a specific voxel and pencil beam index ($i = 568$ and $\alpha = 5$) corresponding to the elementary volume in the position $X = 2$, $Y = 1$, $Z = 5$ cm and to the beamlet with initial kinetic energy/nucleon equal to 160 MeV/u and starting position $X = Y = 2$ cm.

In the following figure 4.3 and 4.4, for each of the three values of simulated beam particle histories $N_{\alpha}$, the 20 different values (one per each run) of the
above described variable $<E>$ and $\sigma_E$ are reported in histograms together with a Gaussian fit.

**Fig. 4.3:** Statistical fluctuations of the average deposited energy $\mu_{i,a}$ for MC runs with $10^3$, $10^4$, $5 \times 10^4$ events (from top to bottom).
Fig. 4.4: Statistical fluctuations of the deposited energy standard deviation $\sigma_E$ for MC runs with $10^3$, $10^4$, $5\times10^4$ events (from top to bottom).

As clearly visible, the values of $\mu$ and $\sigma_E$ are statistically fluctuating and, according to the Central Limit Theorem, their probability distribution approach a Gaussian in the limit of a large number of particle histories.

The resulting values of the Gaussian fit of the $\mu$ and $\sigma_E$ distribution confirm this theorem: the RMS (and the sigma of the fit) of the average deposited energy $\langle E \rangle = \mu$, for example, is equal to:
\[ \sigma_{<E>} = \frac{\sigma_E}{\sqrt{N_{TOT}}} \]  \hspace{1cm} (4.9).

It is self evident that the use of the \( \mu_{\alpha}\) and \( \sigma_{\alpha}\) computed by a MC pre-run of limited statistics needs the tuning of the sub-sample size in order to obtain stable values of these matrix elements.

### 4.3 The pencil beam definition

In the FLUKA Monte Carlo code all “events” or “histories” are initiated by primary particles, which in the simplest case are mono-energetic, mono-directional and start from a single point in space.

For particle sources with more complex distributions in energy, space and direction, the user must write, compile and link a special routine and add a specific card (SOURCE) in the input file.

To simulate a complete treatment field with a 3-dimensional active scanning shaping system, many pencil beams has to be defined, thus an external file (raster.txt) is used to set initial position, finite transversal dimension (spot size) and kinetic energy for each one of them.

This information, in the clinical routine, comes from the output of the TPS dose calculation process and will be used to set up the accelerator controls and the beam line deflection system.

Each pencil beam of the treatment field has been shaped according to a Gaussian lateral spread in the transverse directions \( x \) and \( y \), whose \( \sigma_{x,y} \) is given by the spot size embedded in the raster.txt file.

Moreover, to simulate a realistic scenario, at least one component of a real beam line must be added in the simulation: the passive element that has to be
introduced in the ions path, in order to widen the extension of the Bragg peak, is the so called “Ripple filter”.

The need of this beam line component is due to the fact that the single $^{12}$C beamlet Bragg peak is very narrow and thus, in order to cover the target tumor volume extension in depth with a Spread Out Bragg Peak, an extremely large and unrealistic number of pencil beams with different energies would be otherwise necessary.

A ripple filter has then been modelled: it is built with a 3 mm thick Plexiglas (PMMA) foil with a periodic (1.5 mm pitch) structure of fine grooves [fig. 4.5]. This geometry is similar to the one that is used for this beam line component in the carbon ion therapy facilities actually in operation and that will be implemented at the Italian hadrontherapy center CNAO.

Fig. 4.5: The ripple filter geometry implemented in the FLUKA Monte Carlo simulation [Ref. 4.2].

The effect of the ripple filter is clearly visible in the following figure 4.6 which shows the FLUKA MC simulation resulting Bragg peak spread out for a single $^{12}$C ions pencil beam with initial kinetic energy/nucleon equal to 270 MeV/n:
the extension in depth of the peak (FWHM of a Gaussian fit of the peak maximum region) goes approximately from 2 to 5 mm. This width matches the typical size of the calculation grid used by the treatment planning system and is also of the same order of magnitude of the CT resolution in the longitudinal coordinate $z$.

![Graph](image)

**Fig. 4.6:** The simulated ripple filter effect on the dE/dx profile of a single $^{12}$C pencil beam with initial kinetic energy/nucleon equal to 270 MeV/n.

### 4.4 The simulation geometry set up

In this preliminary development of the Monte Carlo TPS validation tool, a simplified and flexible simulation geometry has been defined.
A water phantom with external dimensions equal to $20 \times 20 \times 30$ cm$^3$, as represented in the following figure 4.7, is the “container”, i.e. the Body, of the interesting treatment region. Inside it a scoring volume has been defined, in order to record all the simulation output data, as a cube of $6.0 \times 6.0 \times 6.0$ cm$^3$ and with voxel dimensions equal to $0.25 \times 0.25 \times 0.5$ cm$^3$. All these dimensions and the position within the main body (X, Y, Z shift) can be easily modified in the user written routines.

**Fig. 4.7:** Schematic drawing, not in scale, of the used simulation geometry set up.

The planning target volume (PTV) has been positioned within the scoring volume and its size has been chosen equal to $2.0 \times 2.0 \times 3.2$ cm$^3$ (*ptv.inc* in Appendix A2) corresponding to a number of 294 voxels; also these dimensions and the position of the PTV can be simply changed in the post-processing code. The ripple filter, already described in the previous paragraph 4.3, has been positioned on the beam line, outside the main body at a distance of 1 m from it.
4.5 The TPS fluence verification

The first TPS validation tool using MC code presented in this thesis is a direct check of the pencil beam particle number $N_\alpha$ given by the TPS, using the $\mu_{i,\alpha}$ and $\sigma_{i,\alpha}$ quantities computed by means of the FLUKA simulation.

In this work the RBE effect on the dose calculation, which is of particular relevance in the case of $^{12}$C ions, will not be considered.

It must be stressed that the introduction of the radiobiological effect, both for the treatment planning system and the Monte Carlo code, implies the use of an externally calculated database that gives the RBE value for each of the ions that, with different energies, are present in the radiation field.

It is assumed that the FLUKA Monte Carlo code will share this database with the analytical TPS program.

Then, the first goal to verify the analytical TPS outcome by the FLUKA code is to calculate the number $N_\alpha$ of histories (i.e. of primary $^{12}$C ions) for each pencil beam of the treatment field which yield, in every voxel of the planning target volume, the prescribed average dose value:

$$D_i^0 = \sum_\alpha N_\alpha \mu_{i,\alpha}.$$  \hspace{1cm} (4.10)

In the reported simulation the value of the average physical dose $D_i^0$ has been fixed equal to 1.6 Gy, corresponding to an average deposited energy $<E>$ equal to $10^{10}$ MeV/cm$^3$ (fluence.inc in Appendix A2).

The calculation has been performed fixing the desired accuracy level, i.e. the relative percentage uncertainty in such a way that:

$$\frac{\sigma_{D_i^0}}{D_i^0} \leq 0.02 = R_i^0$$
as previously stated in paragraph 4.1.
The particle number determination procedure is performed only in a selected sub sample of the irradiated geometry voxels with index \( i \) running from 1 to \( n_{\text{vox}} \text{PTV} \), that is the ones belonging to the predefined region of the planning target volume already described in paragraph 4.4.

Moreover, a choice of all the pencil beams defined in the raster.txt file has been done: only the beamlets with the maximum energy deposition, i.e the Bragg peak, within the PTV has been considered.

The described selection operation and the calculation of the matrix elements \( \mu_{\alpha,\beta} \) and \( \sigma_{\alpha,\beta} \) has been done using the user written program fluence.f, reported in Appendix A3, which acts on the output of the FLUKA Monte Carlo simulation.

The resulting values of the \( x, y \) coordinates of the beam spots are variable between -0.625 cm and 0.875 cm, while the incident kinetic energy/nucleon of the chosen \( ^{12}\text{C} \) pencil beams ranges from 244.5 to 272.5 MeV/n, corresponding to a total number of pencil beams equal to 288.

The task of fluence determination is a function minimization problem over a space of parameters of the function itself.

The multi-parameter object function to be minimized has been defined as a chi-square:

\[
\chi^2(N_{\alpha}) = \sum_{i=1}^{n_{\text{vox}}} \frac{\delta_i^2}{\sigma_{i}^2},
\]

(4.11)

where \( \delta_i \) is the difference between the prescribed and the calculated average dose (or deposited energy) in each voxel of the PTV and \( \sigma_{i} \) are the related uncertainties. That is:
\[ \delta_i = D_i^0 - \sum_\alpha N_\alpha \mu_{i,\alpha} \]

and \( \sigma_\delta = 0.02 D_i^0 \).

The results of the function minimization procedure are just the values of the pencil beams particle number \( N_\alpha \) which yield the prescribed average dose in the planning target volume.

This task can be efficiently solved by the use of the Levenberg–Marquardt algorithm (LMA) [Ref. 4.3, 4.4] that interpolates between the Gauss-Newton algorithm (GNA) and the method of gradient descent.

The open source implementation of the LMA used in this work is that one provided by the Netlib library named minpack [Ref. 4.5]. This package has the constraint that the number of free parameters to be minimized must be less than the number of the voxel contributing to the \( \chi^2 \) function. It must be stressed that this limitation is only due to the used minimization method, and its software implementation, but can be easily overcome by means of more refined (and complex) minimization procedure.

The algorithm acts on a multi-parameter FORTRAN function that is defined and supplied by the user and searches for its minimum with respect to one or more parameters. The user must also supply a set of commands to instruct the package what analysis is wanted and to define the variable parameters, assigning a starting value and upper or lower limits, when necessary, in order to prevent them from taking on unphysical values such as, in this case, a negative value of the particle number.

In this work the Levenberg–Marquardt algorithm has been chosen due to his ability to handle the minimization process of function with a very high number of variable parameters.

Moreover, in order to compare the results, a further minimization test has been done using the MINUIT package [Ref. 4.6], a numerical minimization program.
very widely used in particle physics that was written by Fred James, a CERN (Conseil Européen pour la Recherche Nucléaire) staff physicist, in the 1970s. A customized program `fluence_lib.f` (see Appendix A4) has been written in order to perform all the above described functions (define parameters, execute commands, get the best function value, etc.) directly from FORTRAN through subroutine calls.

The obtained results of the minimization can be verified by the analysis of the resulting values of the relative voxel dose difference

\[
\frac{\delta D}{D} = \frac{D_i^0 - \sum_{\alpha} N_{\alpha} \mu_{i,\alpha}}{D_i^0}
\]

reported in the following figure 4.8.

![Histogram and Gaussian fit of the relative difference between the prescribed and the calculated dose in each voxel of the irradiated PTV.](image)

**Fig. 4.8:** Histogram and Gaussian fit of the relative difference between the prescribed and the calculated dose in each voxel of the irradiated PTV.
As can be seen, the prescribed average dose value is obtained (relative voxel dose difference ~ 0) but the achieved accuracy (~ 6 %) is a bit worse than desired. This can be explained by the limited number of simulated histories used to obtain the matrix elements $\mu_{i,\alpha}$.

To be more quantitative, the simulated number of particles for each pencil beam, $N_{MC} = 10.000$, seems to be a much too reduced sub-sample in comparison with the actual $N_{TPS}^{\alpha}$ values, which are in the order of $10^6$. The obtained values of $N_{TPS}^{\alpha}$ are showed in figure 4.12.

The quality of the results obtained by the direct check of the TPS pencil beam particle number $N_{TPS}^{\alpha}$ can be evaluated in the following figure 4.9 which shows the resulting dose distribution on a transverse section, at average $z$, of the planning target volume. The prescribed average physical dose $D_i^0 = 1.6$ Gy has been achieved (within statistical fluctuation) in the voxels belonging to the PTV volume.
Fig. 4.9: Dose distribution on a transverse section, at average $z$, of the planning target volume. The target value of physical dose was: $D_i^0 = 1.6$ Gy.

4.6 The fluence optimization procedure

It has been already shown that to minimize the statistical accuracy of a Monte Carlo simulation it is necessary to increase the total beam particle number and, therefore, the total computing time. This operation can be avoided only in selected bins where the number of energy releases is very high and the relative uncertainty is consequently low.
In all the other target volume bins, indeed, a fluence optimization represents a possible instrument to achieve the desired accuracy level in a computing time as little as possible.

The method goes as follows: instead of simulating the real $N^\text{TPS}_\alpha$ (or more likely a sub sample $N^\text{TPS}_\alpha \cdot f_{\text{global}}$ with $f_{\text{global}} < 1$) that gives the correct dose distribution in each voxel of the PTV, but with an accuracy $R_i$ that can be higher or lower of the acceptable value $R^0$, the number of simulated ions is $N^\text{MC}_{\alpha}$ but with a “weight” $\frac{1}{f_\alpha} = \frac{N^\text{TPS}_\alpha}{N^\text{MC}_\alpha}$.

In such a way the statistical accuracy $R_i$ is given by the $N^\text{MC}_{\alpha}$ actually simulated, but the average voxel dose is correctly reproduced thanks to the weighting factor $1/f_\alpha$.

Since the computing time is given by the time requested to simulate the $N^\text{MC}_{\alpha}$, a reduction of the total CPU time is possible with respect to the simple method of using a unique $f_{\text{global}}$ equal for all the pencil beams.

The task of particle number optimization with respect to accuracy level represents, as well, a least squares minimization of a multi-parameter object function.

The latter has been defined, as previously (see expression 4.11), as a chi-square where $\delta_i$ is the difference between the calculated relative percentage accuracy of the total average energy distribution in the $i$-th voxel and the desired value $R^0_i$.

Using the Central Limit Theorem, it has been assumed that (see expression 4.9)
\[
\sigma_{<E>}^i = \frac{\sigma_E^i}{\sqrt{N_{\text{TOT}}}}
\]

and, using the above reported expression 4.8 with \( N_{\text{TOT}} = \sum_{\alpha=1}^{N_{\text{penal}}} N_{\alpha} \), the following relation has been obtained:

\[
\frac{\sigma_{<E>}^2}{(\langle E \rangle)^2} = \frac{\sigma_i^2}{N_{\text{TOT}} \cdot \mu_i^2} = \frac{1}{N_{\text{TOT}}^2} \cdot \sum_{\alpha} \frac{N_{\alpha} \cdot (\sigma_{i,\alpha}^2 + \mu_{i,\alpha}^2)}{\mu_i^2} - \frac{1}{N_{\text{TOT}}}.
\]

The object function is consequently given by the expression:

\[
\chi^2(N_\alpha) = \sum_{i=1}^{n_{\text{vox}}^{\text{PTV}}} \frac{\delta_i^2}{\sigma_i^2}
\]

where

\[
\delta_i = \left( \frac{\sigma_{<E>}^i}{\langle E_i \rangle} \right)^2 - \left( R_i^0 \right)^2 = \frac{\sigma_i^2}{N_{\text{TOT}} \cdot \mu_i^2} - \left( R_i^0 \right)^2
\]

and the related uncertainties has been defined as \( \sigma_i = 20\% \cdot R_i^0 \), following the rule of having a 20% tolerance on the resulting values of the voxel sigmas \( \sigma_i \).

As clearly evident in the latter expression, the pencil beam fluences \( N_{\alpha} \) are the only object function parameters that can be varied in order to find the \( \chi^2 \) function minimum, that is to make the difference between the calculated and the desired value of the average energy variance smaller and smaller.

The minimization procedure has been implemented using again the LMA algorithm, with a different user defined object function (see Appendix A4).

This has allowed us to determine the best value of the variable parameters \( N_{\alpha}^{MC} \) which yield the desired relative accuracy \( R^0 = \frac{\sigma_{<E>}}{\langle E \rangle} = 2\% \) [fig. 4.10].
Fig. 4.10: Histogram of the dose relative accuracy values obtained after the particle number minimization procedure in the 294 voxels belonging to the irradiated PTV.

As shown in the previous figure 4.10, the resulting values of the dose relative accuracy have mean equal to 0.02, that is the desired 2% level, and their variation is well contained within the 20% tolerance margin set in the minimization procedure.

Moreover, the performed particle number optimization has allowed to determine the weight factor that has to be applied to the simulated number of $^{12}$C ions $N_{\alpha}^{MC}$ of each pencil beam, in order to obtain the average dose distribution that
is achievable by the simulation of all the particles $N_\alpha^{TPS}$ requested by the analytical treatment planning system.

The resulting values of the weight factor for each of the 288 simulated pencil beams are reported in the following figure 4.11.

![Figure 4.11](image)

**Fig. 4.11:** Weight factors resulting from the fluence optimization procedure.

In the following figure 4.12, the logarithm of the number of carbon ions that each beamlet should provide according to the described TPS fluence verification procedure is reported in black. In red are also reported the logarithm of the same quantities but scaled by an overall weight factor $f_{global}$. This simple scaling allows to generate the minimum sample of $^{12}$C ions to obtain the desired
dose accuracy. Superimposed in blue is then reported the logarithm of the number of carbon ions obtained by the fluence optimization procedure.

![Graph showing dose distribution](image)

**Fig. 4.12:** Number of pencil beam particle used by the TPS and by the Monte Carlo simulation, equally scaled or individually weighted, to obtain the prescribed average dose distribution in the planning target volume.

In addiction, the optimization of the Monte Carlo simulated particle number $N_{\alpha}^{\text{MC}}$ which yields the desired accuracy level $R^0$, together with the determination of the weighting factor $1/f_\alpha$ for each pencil beam of the treatment field, has allowed to achieve a reduction of the overall Monte Carlo simulation computing time by a factor of 2 with respect to the use of a global scaling factor.
Finally it must be added that the obtained results on both the developed procedures, could be improved by an increase of the $N_{\alpha}^{TPS}$ sub sample size, that is of the number $N_{\alpha}^{MC}$ of $^{12}$C ions which are simulated in the pre-run for each pencil beam of the treatment field. Furthermore, a more realistic result could be obtained introducing the condition to have minimal dose (the less the better) outside the tumor region.

Last, but not least, the use of a more refined software tool (i.e. simulated annealing minimization) would permit to deal with a number of pencil beams higher than the number of the PTV voxels. This would then allow the handling of pencil beams with different or reduced spot size, resulting in a better dose conformation to the tumor volume.
CONCLUSIONS AND PERSPECTIVES

This thesis represents a preliminary step in the development of a complete and reliable Monte Carlo package for the validation and optimization of an analytical carbon ion treatment planning system, as foreseen in the INFN - TPS research project.

The use of Monte Carlo codes, such as the chosen FLUKA package, represents indeed the most powerful tool for a precise calculation of dose deposition because it provides a realistic representation of the physical interactions between carbon ions and target material, the detailed transport both of the primary beam and the resulting secondary particles, and because it allows an accurate patient body modelling.

This approach, up to know, has been heavily limited by the huge amount of CPU time requested by the simulation that makes the routine use of this software unfeasible in the daily clinical practice.

However new hints for the use of Monte Carlo codes appeared lately: increasing CPU power is now available by means of computer cluster or GRID distributed computing technology. With the aid of this new hardware is possible to have an accurate evaluation of the dose released in the target volume in a reasonable time (~hour) simulating, with an appropriate weighting factor, only a statistically significant sub sample of the full number of particles actually delivered to the patient.

In this scenario, this thesis studies two independent methods to use the FLUKA Monte Carlo code to validate the analytical optimization of dose delivery. Due to the development stage of these methods, a simplified approach has been chosen where the patient is represented by a water phantom and the radiobiological effect of the beam has been neglected.
The obtained results, reported in Chapter 3 and 4 of this work, show that the choice of a correct simulation set up and the use of a fluence optimization program permit to increase the overall efficiency of the Monte Carlo simulation reducing, at the same time, the processing time.

In Chapter 3 has been shown that a careful choice of the delta rays production and transport kinetic energy threshold translates, without compromising the accuracy of the calculation and with no significant effects on the beam broadening, in a reduction of the total computing time ranging from 4.5 to 6, depending on the primary $^{12}$C ion beam energy.

In Chapter 4 a twofold method has been presented that consists in both direct TPS fluence verification and optimization procedure.

The basic idea is to evaluate by MC, on a statistically significant sub-sample, the average values and the dispersions of the energy release, due to each carbon ion pencil beam, in each voxel of the target volume. On a toy model of a water equivalent planning target volume, using this MC information, it has been extracted:

a) The number of carbon ions to be delivered for each pencil beam to achieve the prescribed dose, thus allowing a direct check by comparison of the analytical TPS results;

b) The number of the carbon ions to be simulated, with the correct weighting factor for each pencil beam of the treatment field, so to achieve an estimation of the delivered dose within the requested statistical accuracy with less CPU time compared with the choice to simulate all the pencil beam fluences scaled by a unique factor.

The results of this technique are very promising: the MC direct computation of the pencil beam fluence (a) has reproduced the desired dose in the target within few %, while the optimization procedure (b) has produced a further reduction...
by a factor of 2 of the total CPU time with respect to the case of applying the same global weighting factor to all the pencil beams.

On the other hand, the work presented can be extended in several directions, some of which are:

- A more refined optimization software than the LM used can be introduced, allowing more freedom in the ratio requested between the number of treated voxels and the number of used pencil beams (i.e. simulated annealing, etc.);
- The accurate description of a realistic target can be achieved by the inclusion in the MC simulation input file of the density and atomic composition info translated from the patient diagnostic CT images;
- The optimization procedure can be performed in terms of the isoeffective dose including the radiobiological effect info by means of an externally calculated database of RBE values for each ion present in the radiation field.

The implementation of these items is an interesting challenge for future work that, together with the yet described optimization tools, will allow the development of a complete and reliable FLUKA Monte Carlo package for the validation and optimization of an analytical carbon ion treatment planning system to be used in the clinical practice.
REFERENCES


[1.3] H.A. Bethe, Annals of Physics 5, 1930, p. 325-400


[1.24] [http://ptcog.web.psi.ch/ptcentres.html](http://ptcog.web.psi.ch/ptcentres.html)

[1.25] [http://ptcog.web.psi.ch/newptcentres.html](http://ptcog.web.psi.ch/newptcentres.html)


INTERNATIONAL COMMISSION ON RADIATION UNITS AND MEASUREMENTS, *Tissue substitutes in radiation dosimetry and measurement* - ICRU Report 44, 1989, Bethesda, MD

http://wwwasd.web.cern.ch/wwwasd/paw/


F. Bourhaleb et al., *Monte Carlo simulations of ripple filters designed for proton and carbon ion beams in hadrontherapy with active scanning technique*, Journal of Physics: Conference Series 102, 2008


http://www.netlib.org/minpack

ACKNOWLEDGEMENTS

I would like to sincerely thank all the people without whom this diploma thesis work would never be accomplished.

First and foremost my supervisors: prof. Vincenzo Patera and prof. Maurizio Pelliccioni. The first for sharing with me his expertise and research insight, for his great patience when I was discouraged and for giving me a continuous boost to go on and do my best.

I want to express my gratitude to Prof. Pelliccioni for the accurate check of this thesis he made and for his precious suggestions and remarks.

I would like also to thank some people of the FLUKA collaboration group for their kindly help and useful scientific advice: Cristina Morone, Anna Ferrari, Andrea Mairani, Paola Sala and Giuseppe Battistoni.

Very special thanks go to Paolo, my love, for the priceless aid and the endless support and encouragement he gave me during the last year.

I am grateful also to my SSFS journey companions: Michela, Filippo, Tatiana and Antonio for sharing with me their study experience and for their friendship.

Last, but not least, I would like to thank my family for love, patience and support during the last four years and for giving me the chance to become a Medical Physicist.
## APPENDIX

### [A1] Hadrontherapy Patients Statistics

**Facilities out of operation:**

<table>
<thead>
<tr>
<th>WHERE</th>
<th>WHAT</th>
<th>FIRST PATIENT</th>
<th>LAST PATIENT</th>
<th>PATIENT TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgium</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Canada</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Russia</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sweden</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switzerland</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CA, USA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IN, USA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NM, USA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Facilities in operation:**

<table>
<thead>
<tr>
<th>WHERE</th>
<th>WHAT</th>
<th>FIRST PATIENT</th>
<th>PATIENT TOTAL</th>
<th>DATE OF TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Canada</td>
<td></td>
<td>1990</td>
<td>1380</td>
<td>Dec-07</td>
</tr>
<tr>
<td>China</td>
<td></td>
<td>2004</td>
<td>537</td>
<td>Dec-07</td>
</tr>
<tr>
<td>England</td>
<td></td>
<td>1969</td>
<td>1701</td>
<td>Dec-07</td>
</tr>
<tr>
<td>France</td>
<td></td>
<td>1981</td>
<td>3129</td>
<td>Sep-06</td>
</tr>
<tr>
<td>France</td>
<td></td>
<td>1991</td>
<td>4143</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Germany</td>
<td></td>
<td>1967</td>
<td>394</td>
<td>Oct-07</td>
</tr>
<tr>
<td>Germany</td>
<td></td>
<td>1998</td>
<td>1014</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Italy</td>
<td></td>
<td>2002</td>
<td>151</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td>1994</td>
<td>3795</td>
<td>Jan-06</td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td>1998</td>
<td>522</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td>2001</td>
<td>1658</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td>2001</td>
<td>271</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td>2001</td>
<td>188</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td>2002</td>
<td>49</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Japan</td>
<td></td>
<td>2003</td>
<td>570</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Korea</td>
<td></td>
<td>2007</td>
<td>155</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Russia</td>
<td></td>
<td>1969</td>
<td>4034</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Sweden</td>
<td></td>
<td>1975</td>
<td>1327</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Switzerland</td>
<td></td>
<td>1969</td>
<td>402</td>
<td>Dec-07</td>
</tr>
<tr>
<td>Switzerland</td>
<td></td>
<td>1994</td>
<td>520</td>
<td>Mar-07</td>
</tr>
<tr>
<td>Switzerland</td>
<td></td>
<td>1990</td>
<td>12689</td>
<td>Mar-08</td>
</tr>
<tr>
<td>IN, USA</td>
<td></td>
<td>2004</td>
<td>379</td>
<td>Dec-07</td>
</tr>
<tr>
<td>MA, USA</td>
<td></td>
<td>2001</td>
<td>2710</td>
<td>Oct-07</td>
</tr>
<tr>
<td>TX, USA</td>
<td></td>
<td>2006</td>
<td>527</td>
<td>Jan-08</td>
</tr>
<tr>
<td>FL, USA</td>
<td></td>
<td>2006</td>
<td>360</td>
<td>Dec-07</td>
</tr>
</tbody>
</table>

Total for all facilities (in operation and out of operation):

- 50130 Grand Total
- 2054 He
- 11403 protons
- 4450 C-ions
- 433 other ions
- 55983 protons

---

89
The include files: ptv.inc and fluence.inc

c Include file ptv.inc
c
c dimensioni dei voxel CT
c
real vox_dx, vox_dy, vox_dz
parameter (vox_dx=0.25, vox_dy=0.25, vox_dz=0.5)
c
c definizione parametri zona PTV
c
real xmax_PTV, xmin_PTV, ymax_PTV, ymin_PTV, zmax_PTV, zmin_PTV
parameter (xmax_PTV=1.0, xmin_PTV=-1.0)
parameter (ymax_PTV=1.0, ymin_PTV=-1.0)
parameter (zmax_PTV=14.3, zmin_PTV=11.1)
c
c definizione parametri zona CTV
c
real xmax_CTV, xmin_CTV, ymax_CTV, ymin_CTV, zmax_CTV, zmin_CTV
parameter (xmax_CTV=1.0, xmin_CTV=-1.0)
parameter (ymax_CTV=1.0, ymin_CTV=-1.0)
parameter (zmax_CTV=14.3, zmin_CTV=11.1)
c
c step dei fasci
c
real pen_dx, pen_dy, pen_dE
parameter (pen_dx=0.3, pen_dy=0.3, pen_dE=0.004)
c
c dimensioni trasverse dei fasci
c
real pen_trasv_size
parameter (pen_trasv_size=0.3)
c Include file fluence.inc

c    integer NBINMAX, NPENMAX
    parameter (NBINMAX=8000, NPENMAX=1500)
    integer NUMINFO
    parameter (NUMINFO=12)

c    integer NSCIAMI
    parameter (nsciami = 10000)
    double precision dtarget, sigma_perc_flux, sigma_rel_var
    parameter (dtarget =1.0d10, sigma_perc_flux=0.02)
    parameter (sigma_rel_var=0.02)

c    real   xvox(NBINMAX), yvox(NBINMAX), zvox(NBINMAX)
    double precision dose0(NBINMAX), flusso(NPENMAX)
    double precision flux0(NPENMAX), ene0(NBINMAX)
    double precision flux_sig(NPENMAX)
    double precision MU_AI_0(NBINMAX, NPENMAX)
    double precision MU_AI(NBINMAX, NPENMAX)
    double precision MU_AI_TOT(NBINMAX, NPENMAX)
    double precision SI_AI(NBINMAX, NPENMAX)
    double precision enetot0(NBINMAX)
    integer idvox_CTV(NBINMAX), idvox_PTV(NBINMAX)
    integer idvox_TOT(NBINMAX)
    integer ipenok(NPENMAX)
    integer nvox_CTV, nvox_PTV, npenok, nvox_tot
    common /fluence_info/MU_AI, SI_AI, MU_AI_0, MU_AI_TOT, 
        $ dose0, flusso, flux0, 
        $ enetot0, ene0, flux_sig,xvox, yvox, zvox, 
        $ nvox_CTV, nvox_PTV, npenok, nvox_tot, 
        $ idvox_CTV, idvox_PTV, idvox_TOT, ipenok

c    integer aprohis, chiudohis
    parameter (aprohis=0, chiudohis=1)
    integer knoweight, kweight
    parameter (knoweight=0, kweight=1)
[A3] The fluence.f program

program fluence
*
implicit none
include 'fluence.inc'
include 'ptv.inc'
c
integer hmemor(5000000),iquest(100)
common/pawc/hmemor
common/quest/iquest
c
Integer istat,ii,kk,jj
character chrout
real ave_sig_media,amax_sig_media,amin_sig_media
real drel,ave_drel_dose,zave
tot_sciami_raw,tot_sciami_sig,alog_flux_sig
double precision dose(NBINMAX)
c
call hlimit(5000000)
c
scelgo i voxel del CTV e i pencil da trattare
c
IQUEST(10) = 65000
call HROPEN(5,'fluhis','fluence.his',' ',4096,istat)
if(istat.ne.0) go to 2000
call hrin(1,9999,0)
call HGNPAR(1,CHRROUT)
c
write(6,123)
c
call scelgo_CTV()
c
trovo i pencil beam attivi

call scelgo_pencil()
c
write(6,234) nvox_CTV, npenok
c
trovo la matrice dei rilasci dei pencil beam sui voxel del CTV
c
call creo_matrice()
c
call hrend('fluhis')
close(5)
c
call histogrammi(srohis)
c
calibrate the starting flux so to give the correct averaged dose
c
ave_drel_dose = 0
do kk = 1,nvox_ctv
dose(kk) = 0
do ii = 1, npenok
dose(kk) = dose(kk) + mu_ai(kk,ii)*flux0(ii)
end do
ave_drel_dose = ave_drel_dose + dose(kk)/dose0(kk)
end do
ave_drel_dose = ave_drel_dose/float(nvox_ctv)
c
write(6,'(a)')'starting fluxes:'
do ii = 1, npenok
flux0(ii) = flux0(ii)/ave_drel_dose
write(6,*)'pencil # ',ii,'  flux0= ',flux0(ii)
end do

start of minimization

call find_fluxes_lm()

verify the fluxes output

write(6,*)' flusso minimizzato:'
do kk = 1,nvox_cv
  dose(kk) = 0
  do ii = 1, npenok
    dose(kk) = dose(kk) + mu_ai(kk,ii)*flusso(ii)
  end do
  drel = (dose0(kk)-dose(kk))/dose0(kk)
  write(6,*)'voxel= ',kk,' dose0= ',dose0(kk),' dose_out= ',
  & dose(kk),', drel = ',drel
end do

start sigma minimization: find a scale factor

call verifica_sigma(knoweight, ave_sig_media, amax_sig_media,
  & amin_sig_media)
tot_sciami_raw = 0
  do ii = 1, npenok
    flux_sig(ii) = flusso(ii)*(ave_sig_media/sigma_rel_var)**2
    tot_sciami_raw = tot_sciami_raw + flux0(ii)
      if(flux0(ii).gt.0) then
        alog_flux = sngl(log10(flusso(ii)*
          & ave_sig_media/sigma_rel_var)**2))
        call hf1(i4,alog_flux,1.)
      endif
  end do
  tot_sciami_raw = tot_sciami_raw*(ave_sig_media/sigma_rel_var)**2

call mini_sigma_minuit()
call mini_sigma_lm()

call verifica_sigma(kweight, ave_sig_media, amax_sig_media,
  & amin_sig_media)
tot_sciami_sig = 0
  do ii = 1, npenok
    tot_sciami_sig = tot_sciami_sig +flux0(ii)
      if(flux0(ii).gt.0) then
        call hf1(i5,sngl(log10(flux0(ii))),1.)
      endif
  end do
  write(6,1111)tot_sciami_raw, tot_sciami_sig,
  & tot_sciami_sig/tot_sciami_raw

call verifica_sigma(kweight, ave_sig_media, amax_sig_media,
  & amin_sig_media)
tot_sciami_sig = tot_sciami_sig/tot_sciami_raw

call histogrammi(chiudohis)
2000 continue
1111 format(/,'# sciami per sigma ok: RAW=',2x,f12.1,2x,
$     'POST mini_sigma=',2x,f12.1,2x,'ratio=',2x,f7.4)
123 format(/,'----------- PROGRAM FLUENCE -----------',/)
234 format(/,'nvox_CTV=',1x,I4,3x,'npenok=',1x,I4,/,)
654 format(/,'RISULTATI TPS, MINIMIZZAZIONE, WEIGHT:',/)
876 format('pencil #',1x,I3,2x,'Ntps=',1x,i10,2x,'Nmc=',1x,I8,2x,
$     'Weight=',1x,f8.3)
756 format(/,'risultati minimizzazione dose:')
c
stop
c end

c *********** SUBROUTINES ***********
-----------------------------------------------------------
C
C subroutine histogrammi(iflag)
-----------------------------------------------------------
C
C IMPLICIT NONE
integer istat, iflag, icicle
include 'fluence.inc'
include 'invert_std.inc'
c
if(iflag.eq.aprohis) then
  call HROPEN(5,'fluhis','fluence_res.his','N',1024,istat)
c  call hbook1(10,'relative voxel dose difference ',
$    100,-1.,1.,0.)
  call hbook2(11,'dose x vs y integrated along z',
$    nbinx,xshift,-xshift,nbiny,yshift,-yshift,0.)
  call hbook2(12,'dose x vs y at average z',
$    nbinx,xshift,-xshift,nbiny,yshift,-yshift,0.)
  call hbook2(21,'dose x vs z integrated along y',
$    nbinx,xshift,-xshift,nbinz,zshift,zshift+nbinz*dz,0.)
  call hbook2(22,'dose x vs z at average y',
$    nbinx,xshift,-xshift,nbinz,zshift,zshift+nbinz*dz,0.)
  call hbook2(31,'delta dose x vs y integrated along z',
$    nbinx,xshift,-xshift,nbiny,yshift,-yshift,0.)
  call hbook2(32,'delta dose x vs z integrated along y',
$    nbinx,xshift,-xshift,nbiny,yshift,-yshift,0.)
  call hbook2(201, 'Dose relative accuracy after fluence minimization',
$    1000,0.0,0.04,0.)
  call hbook2(202,'Sigma average dose RAW (%)',
$    1000,0.0,0.04,0.)
  call hbook1(13,'log of pencil beam particle number ',100,
$    2.0,8.0,0.)
  call hbook1(14,'log of downscaled pencil beam particle number','
$    100,2.0,8.0,0.)
  call hbook1(15, 'log of optimized pencil beam particle number',
$    100,2.0,8.0,0.)
  call hbook1(16,'Weight ',1000,0.0,1000.0,0.)
elseif(iflag.eq.chiudohis) then
  call hrout(0,icicle,'t')
close(5)
cendif
C
return
c end
c------------------------------------------------------------
subroutine analisi()
c------------------------------------------------------------

implicit none
include 'fluence.inc'
real dose(NBINMAX)
c
integer kk, ii, idv
real zave, yave, drel(NBINMAX)
parameter (zave=12.5, yave=0.0)
c
do kk = 1, nvox_ctv
  dose(kk) = 0
  idv = idvox_ctv(kk)
do ii = 1, npenok
  dose(kk) = dose(kk) + mu_ai(kk,ii)*flusso(ii)
end do
drel(kk) = (dose0(kk)-dose(kk))/dose0(kk)
call hf1(10, drel(kk), 1.)
call hf2(31, xvox(idv), yvox(idv), abs(drel(kk)))
call hf2(32, xvox(idv), zvox(idv), abs(drel(kk)))
end do
c
do ii = 1, npenok
  if (flusso(ii).gt.0) then
call hf1(13, sngl(log10(flusso(ii))), 1.)
  endif
end do
c
do kk = 1, nvox_tot
  dose(kk) = 0
  idv = idvox_tot(kk)
do ii = 1, npenok
  dose(kk) = dose(kk) + mu_ai_tot(idv,ii)*flusso(ii)
end do
call hf2(11, xvox(idv), yvox(idv), dose(kk)*1.602/1.e10)
call hf2(21, xvox(idv), zvox(idv), dose(kk)*1.602/1.e10)
  if (zvox(idv).eq.zave) then
call hf2(12, xvox(idv), yvox(idv), dose(kk)*1.602/1.e10)
  endif
if (yvox(idv).eq.yave) then
call hf2(22, xvox(idv), zvox(idv), dose(kk)*1.602/1.e10)
  endif
end do
c
return
end

c------------------------------------------------------------
subroutine scelgo_CTV()
c------------------------------------------------------------

c  scelgo il CTV

implicit none
include 'fluence.inc'
real xvect(NUMINFO)
c
integer idvoxel, idpen, nevpen
real xx, yy, zz, aedep, avesq, sige, xpen, ypen, ekpen
c
integer kk, ii, nevent, ierr, iflag_ptv
logical new_voxel
integer PTV_domain
external PTV_domain
selection region

write(6,3111)

nvox_CTV = 0
call vzero(idvox_CTV,NBINMAX)
call vzero(ene0,2*NBINMAX)
call vzero(enetot0,2*NBINMAX)
nvox_PTV = 0
call vzero(idvox_PTV,NBINMAX)
call vzero(dose0,2*NBINMAX)
call vzero(xvox,NBINMAX)
call vzero(yvox,NBINMAX)
call vzero(zvox,NBINMAX)
nvox_tot = 0

CALL HNOENT(1,NEVENT)

do kk = 1,nevent
CALL HGNF(1,kk,xvect,IERR)
   idvoxel = int(xvect(1))
   xx = xvect(2)
   yy = xvect(3)
   zz = xvect(4)
   avedep = xvect(5)
   avesq = xvect(6)
   sige = xvect(7)
   xpen = xvect(8)
   ypen = xvect(9)
   ekpen = xvect(10)
   nevpen = int(xvect(11))
   idpen = int(xvect(12))
   if(idpen.eq.0) then
      nvox_tot = nvox_tot + 1
      xvox(idvoxel) = xx
      y vox(idvoxel) = yy
      zvox(idvoxel) = zz
      idvox tot(nvox_tot) = idvoxel
   endif
   iflag_PTV = PTV_domain(xx,yy,zz)
   if((nv ox_CTV.lt.NBINMAX) .and. (iflag_PTV.eq.1)) then
      write(6,*)nvox_tot,xx,yy,zz,idvoxel
   endif
voxel is in CTV .. already flagged?
ii = 1
new_voxel = .true.
do while((ii.le.nvox_CTV).and.(new_voxel))
   if(idvoxel.eq.idvox_CTV(ii)) then
      new_voxel = .false.
   endif
   ii = ii + 1
end do
ok is a new good voxel
if(new_voxel) then
   nvox_CTV = nvox_CTV + 1
   idvox_CTV(nvox_CTV) = idvoxel
endif
if is in PTV high dose, if in (CTV-PTV) dose = 0

97
if(iflag_ptv.eq.1) then
    dose0(nvox_CTV) = dtarget
    nvox_PTV = nvox_PTV + 1
    idvox_PTV(nvox_PTV) = idvoxel
    write(6,1111)nvox_CTV,idvox_CTV(nvox_CTV),xx,yy,zz
else
    dose0(nvox_CTV) = 0
endif
endif
endif
end do

3111 format('Selection of the CTV voxels:',/)
1111 format('vox # ',1x,I4,1x,'id=',1x,I4,2x,'x,y,z=',3(1x,f6.3))
return
end

c-------------------------------------------------------------
subroutine scelgo_pencil()
-------------------------------------------------------------
 NB : seleziona i pencils che hanno il max in PTV

implicit none
include 'fluence.inc'
real xvect(NUMINFO)
INTEGER idvoxel, idpen, nevpen
REAL xx, yy, zz, avedep, avesq, sige, xpen, ypen, ekpen
INTEGER ii, ierr, kk, idpen_previous, nloop, ivox_emax
logical flag_new_pencil, flag_in_PTV
real emax, flusso_iniziale
real xpen_previous, ypen_previous, ekpen_previous
write(6,4111)
c
npenok = 0
call vzero(ipenok,NPENMAX)
cALL HNOENT(1,NLOOP)
write(6,*)'nloop= ', nloop
idpen_previous = 0
emax = 0
ivox_emax = 0
do kk = 1,nloop
    call HGNF(1,kk,xvect,IERR)
    idvoxel = int(xvect(1))
    xx = xvect(2)
    yy = xvect(3)
    zz = xvect(4)
    avedep = xvect(5)
    avesq = xvect(6)
    sige = xvect(7)
    xpen = xvect(8)
    ypen= xvect(9)
    ekpen = xvect(10)
    nevpen = int(xvect(11))
    idpen = int(xvect(12))
c
c analize info on the previous pencil beam bragg peak

c
if((idpen_previous.ne.idpen).and.(kk.ne.1).and. 
(idpen.ne.0)) then ! changed pencil 

write(6,'(changed pencil-> id_previous ',idpen_previous, 
', vox max = ', ivox_emax,' emax=',emax

flag_in_PTV = .false.
ii = 1 
do while((ii.le.nvox_PTV).and.(.not.flag_in_PTV)) 
if(ivox_emax.eq.idvox_PTV(ii)) then 
flusso_iniziale = dose0(ii)/emax 
flag_in_PTV = .true.
endif 
ii = ii + 1 
enddo

if(flag_in_PTV) then 
flag_new_pencil = .true.
ii = 1 
do while((ii.le.npenok).and.(flag_new_pencil)) 
if(ipenok(ii).eq.idpen_previous) then 
flag_new_pencil = .false.
endif 
ii = ii +1 
end do

if(avedep.gt.emax) then 
emax = avedep 
ivox_emax = idvoxel 
endif

end if

if(avedep.gt.emax) then 
end do

4111 format(/,'Selection of the pencils to be minimized:',/)
4444 format('pencil #',1x,I3,4x,'id=',1x,I4,4x,'Emax=',1x,f12.5, 
3x,'flux0=',1x,f12.2)
5555 format(4x,'x,y beam=',1x,f7.3,1x,f7.3, 
3x,'E(MeV/u)=',1x,f7.3,2x,'idvox=',1x,i4)

return
end
c-------------------------------------------------------------
subroutine creo_matrice()
-------------------------------------------------------------
c implicit none
include 'fluence.inc'
real xvect(NUMINFO)
c
INTEGER idvoxel, idpen, nevpen
REAL xx, yy, zz, avedep, avesq, sige, xpen, ypen, ekpen
c
INTEGER ii, nloop, kk, ierr, iorder_pen_ok
logical flag_vox_ok, flag_pencil_ok
c
CALL HNOENT(1,NLOOP)
call vzero(mu_ai,2*NPNMAX*NBINMAX)
call vzero(mu_ai_0,2*NPNMAX*NBINMAX)
call vzero(mu_ai_tot,2*NPNMAX*NBINMAX)
call vzero(si_ai,2*NPNMAX*NBINMAX)
c
do kk = 1,nloop
    CALL HGNF(1,kk,xvect,IERR)
idvoxel = int(xvect(1))
    xx = xvect(2)
    yy = xvect(3)
    zz = xvect(4)
    avedep = xvect(5)
    avesq = xvect(6)
    sige = xvect(7)
    xpen = xvect(8)
    ypen = xvect(9)
    ekpen = xvect(10)
    nevpen = int(xvect(11))
idpen = int(xvect(12))
    * seleziono i pencil beams

    if(ipenok(gt.0)) then
        ii = 0
        do while((.not.flag_pencil_ok).and.(ii.le.npenok))
            ii = ii + 1
            if(ipenok(ii).eq.idpen) then
                flag_pencil_ok = .true.
                iorder_pen_ok = ii
            endif
        end do

    endif

c    ok the pencil beam is good.. then look at the voxel

c    if(flag_pencil_ok) then
        if(idvoxel.le.nbinmax) then
            mu_ai_tot(idvoxel,iorder_pen_ok) = avedep
        endif
        flag_vox_ok = .false.
        ii = 0
        do while((.not.flag_vox_ok).and.(ii.le.nvox_CTV))
            ii = ii + 1
            if(idvox_CTV(ii).eq.idvoxel) then
                if((sige.gt.0).or.(avedep.gt.0)) then
                    write(6,432)ii,iorder_pen_ok,sige,avedep
                endif
            endif
        end do
    endif

    write(6,432)ii,iorder_pen_ok,sige,avedep
    mu_ai(ii,iorder_pen_ok) = dble(avedep)

100
mu_ai_0(ii,iorder_pen_ok) = dble(avedep)
si_ai(ii,iorder_pen_ok) = dble(sige)
c
$write(6,*)'j=',ii,' alpha=',iorder_pen_ok,
$mu(a,i)= ',mu_ai(ii,iorder_pen_ok)
endif
c
do ii=1,nvox_CTV
do iorder_pen_ok=1,npenok
mu_ai(ii,iorder_pen_ok) = dble(avedep)
si_ai(ii,iorder_pen_ok) = dble(sige)
c
$write(6,*)'j=',ii,' alpha=',iorder_pen_ok,
$mu(a,i)= ',mu_ai(ii,iorder_pen_ok)
endif
do
endif
do
c
c c------------------------------------------------------------------
csubroutine verifica_sigma(iswitch, ave_sig_media,$
$ amax_sig_media, amin_sig_media)
c------------------------------------------------------------------
cc compute the value of the sigma for each voxels and compare it to$ cthe goal value
cc implicit none
cc include 'fluence.inc'
cc INTEGER ii, ialpha, iswitch
double precision sigma2(NBINMAX),amedia(NBINMAX)$
real sig_media, ave_SIG_MEDIA, amin_sig_media, amax_sig_media
c
call vzero(amedia,2*NBINMAX)
call vzero(sigma2,2*NBINMAX)$
ave_sig_media = 0
amin_sig_media = 100000.
amax_sig_media = 0
c
c write(6,111)
c
c if(iswitch.eq.knoweight) then
do ialpha=1,npenok
enne(ialpha) = flusso(ialpha)
do
elseif(iswitch.eq.kweight) then
do ialpha=1,npenok
enne(ialpha) = flux_sig(ialpha)
do
endif
c
enne_tot = 0
do ialpha=1,npenok
enne_tot = enne_tot + enne(ialpha)
do
endif
c
if(enne_tot.le.0) goto 1000
c
do ii=1,nvox_CTV$eleded
$do iorder_pen_ok=1,npenok
elmat = si_ai(ii,ialpha)*si_ai(ii,ialpha) +
$mu_ai(ii,ialpha)*mu_ai(ii,ialpha)$
sigma2(ii) = sigma2(ii) + elmat*enne(ialpha)
do
sigma2(ii) = sigma2(ii)/enne_tot

432 $format('Crea_matrice -> ERROR !! matrix < 0: ii=',1x,i3,$
$ 2x,'ialpha=',1x,i3,2x,'sige=',1x,f9.4,2x,'avedep=',',1x,f9.4)$
return
c
end
aus = 0
do ialpha = 1,npenok
   aus = aus + enne(ialpha)*mu_ai(ii,ialpha)
end do
amedia(ii) = aus/enne_tot
sigma2(ii) = (sigma2(ii) - amedia(ii)*amedia(ii))/enne_tot

write(6,479)ii,dsqrt(sigma2(ii))/amedia(ii)
sig_media = dsqrt(sigma2(ii))/amedia(ii)
if(iswitch.eq.kweight) then
   call hf1(201,sig_media,1.)
elseif(iswitch.eq.knweight) then
   call hf1(202,sig_media,1.)
endif
if(sig_media.gt.amax_sig_media) amax_sig_media = sig_media
if(sig_media.lt.amin_sig_media) amin_sig_media = sig_media
ave_sig_media = ave_sig_media + dsqrt(sigma2(ii))/amedia(ii)
end do
ave_sig_media = ave_sig_media/float(nvox_CTV)

1000 continue

111  format(/,' verifica sigma:',/)
479  format('idvox=',1x,i3,2x,'sigma/dose media=',1x,f7.5)

return
end
[A4] The fluence_lib.f program

```fortran
integer function PTV_domain(xx,yy,zz)  
c------------------------------------------------------------------  
include 'ptv.inc'  
real xx,yy,zz  
PTV_domain = 0  
$  (zz.lt.zmax_CTV).and.(zz.gt.zmin_CTV) ) then  
  PTV_domain = 1  
endif  
return  
end

c------------------------------------------------------------------  
real function E_BPeak(zz)  
c------------------------------------------------------------------  
fit to the kinetic energy (MeV/nucl) of Carbon 12 needed to have  
the B.P. at zz (cm)  
real espo, fattore  
parameter ( fattore = 0.0008, espo = 0.57283611)  
E_BPeak = (zz/fattore)**espo  
return  
end

c------------------------------------------------------------------  
subroutine find_fluxes_lm()  
c------------------------------------------------------------------  
implicit none  
include 'fluence.inc'  
INTEGER LDFJAC,INFO,LWA  
INTEGER IPVT(NPENMAX)  
DOUBLE PRECISION TOL,FNORM  
DOUBLE PRECISION XVAL(NPENMAX),FVEC(NBINMAX),FJAC(NBINMAX,NPENMAX)  
DOUBLE PRECISION WA(5*NBINMAX*NPENMAX)  
EXTERNAL FCNLIM  
integer ii  
double precision scale_flux, scale_sigma  
common /lm_common/scale_flux, scale_sigma  
C  
C LOGICAL OUTPUT UNIT IS ASSUMED TO BE NUMBER 6.  
C  
write(6,444)  
LDFJAC = nvox_CTV  
LWA = 5*NBINMAX*NPENMAX  
C
```
SET TOL TO THE SQUARE ROOT OF THE MACHINE PRECISION.
UNLESS HIGH PRECISION SOLUTIONS ARE REQUIRED,
THIS IS THE RECOMMENDED SETTING.

TOL = DSQRT(DPMPAR(1))

starting value of the parameters

scale_flux = 1.0d4

do ii = 1,npenok
    xval(ii) = flux0(ii)/scale_flux
end do

CALL LMDER1(FCNLM,nvox_CTV,npenok,XVAL,FVEC,FJAC,LDFJAC,TOL,
            INFO,IPVT,WA,LWA)
FNORM = ENORM(nvox_CTV,FVEC)

do ii=1,npenok
    flusso(ii) = xval(ii)*scale_flux
    write(6,555)ii,xval(ii)
end do

444  format(/,'LM minimization of fluxes:/')
555  format('par #',1x,i3,2x,'=',1x,f8.3)

return
end

c-------------------------------------------------------
SUBROUTINE FCNLM(M,N,XVAL,FVEC,FJAC,LDFJAC,IFLAG)
c-------------------------------------------------------
implicit none
include 'fluence.inc'
INTEGER M, N, LDFJAC, IFLAG
DOUBLE PRECISION XVAL(N), FVEC(M), FJAC(LDFJAC,N)
integer ii, ialpha
double precision somma, delta, enne_negativi
logical nalpha_negativo(NPENMAX)
logical un_nalpha_negativo

scale_flux, scale_sigma
common /lm_common/scale_flux, scale_sigma

c delta/sig_delta vector calculation
IF (IFLAG.eq.1) then
check on the positive value of the fluxes
enne_negativi = 0.
un_nalpha_negativo = .false.
do ialpha = 1, npenok
    if(xval(ialpha).gt.0) then
        nalpha_negativo(ialpha) = .false.
    else
        nalpha_negativo(ialpha) = .true.
        un_nalpha_negativo = .true.
        enne_negativi = enne_negativi + xval(ialpha)
    endif
end do

if(.not.un_nalpha_negativo) then
somma = 0.0d0
do ialpha = 1, npenok
  somma = somma +
  xval(ialpha)*MU_AI(ii,ialpha)*scale_flux
end do
delta = dose0(ii) - somma
if(.not.nalpha_negativo(ialpha)) then
  FJAC(ii,ialpha) = -MU_AI(ii,ialpha)*scale_flux/
                 dose0(ii)/sigma_perc_flux
else
  FJAC(ii,ialpha) = 1.d2*xval(ialpha)*xval(ialpha)
endif
end do
RETURN
END

c------------------------------------------------------------------
subroutine mini_sigma_lm()
------------------------------------------------------------------
implicit none
include 'fluence.inc'

INTEGER LDFJAC,INFO,LWA
INTEGER IPVT(NPENMAX)
DOUBLE PRECISION TOL,FNORM
DOUBLE PRECISION XVAL(NPENMAX),FVEC(NBINMAX),FJAC(NBINMAX,NPENMAX)
DOUBLE PRECISION WA(5*NBINMAX*NPENMAX)
EXTERNAL FCNSIGLM
integer ii,il

double precision scale_flux, scale_sigma
common /lm_common/scale_flux, scale_sigma
C
C     LOGICAL OUTPUT UNIT IS ASSUMED TO BE NUMBER 6.
C
write(6,444)
LDFJAC = nvox_CTV
LWA = 5*NBINMAX*NPENMAX
C
C     SET TOL TO THE SQUARE ROOT OF THE MACHINE PRECISION.
C     UNLESS HIGH PRECISION SOLUTIONS ARE REQUIRED, THIS IS THE RECOMMENDED SETTING.
C
TOL = DSQRT(DPMPAR(1))
C
C     starting value of the parameters
scale_sigma = 1.0d3
do ii = 1,npenok
  xval(ii) = flux_sig(ii)/scale_sigma
end do

CALL LMDER1(FCNSIGLM,nvox_CTV,npenok,XVAL,FVEC,FJAC,LDFJAC,TOL, *
  INFO,IPVT,WA,LWA)
FNORM = ENORM(nvox_CTV,FVEC)

do ii=1,npenok
  if(xval(ii).lt.0) xval(ii) = .1
end do

CALL LMDER1(FCNSIGLM,nvox_CTV,npenok,XVAL,FVEC,FJAC,LDFJAC,TOL, *
  INFO,IPVT,WA,LWA)
FNORM = ENORM(nvox_CTV,FVEC)

write(6,333)fnorm
do ii=1,npenok
  flux_sig(ii) = xval(ii)*scale_sigma
  write(6,555)ii,xval(ii)
end do

Cc 333 format('End sigma minimization: fnorm=',1x,f10.5)
444 format('/','LM minimization for sigma:')/
555 format('par #',1x,i3,2x,'=',1x,f8.3)

return
end

-----------------------------------------
SUBROUTINE FCNSIGLM(M,N,XVAL,FVEC,FJAC,LDFJAC,IFLAG)
-----------------------------------------

implicit none
include 'fluence.inc'

double precision scale_flux, scale_sigma
common /lm_common/scale_flux, scale_sigma

INTEGER M, N, LDFJAC, IFLAG
DOUBLE PRECISION XVAL(N), FVEC(M), FJAC(LDFJAC,N)
integer ii, ialpha
double precision enne_tot, ennemu(NBINMAX), ennewu(NBINMAX)
double precision aus1, aus2, radice(NBINMAX)
logical nalpha_negativo(NPENMAX)
logical un_nalpha_negativo

delta/sig_delta vector calculation
c
if(iflag.eq.1) then
c
check all the Nalpha are positive
c
un_nalpha_negativo = .false.
xval_neg = 0.d0
do ialpha = 1, npenok
  if(xval(ialpha).gt.0) then
    nalpha_negativo(ialpha) = .false.
  else
    nalpha_negativo(ialpha) = .true.
    un_nalpha_negativo = .true.
  end if
end do

return
end
ENNE_TOT = 0
DO IALPHA=1,N
   ENNE_TOT = ENNE_TOT + XVAL(IALPHA)*SCALE_SIGMA
END DO

DO II=1,NVOX_CTV
   ENNEMU(II) = 0
   ENNEWU(II) = 0
   DO IALPHA=1,NPENOK
      ENNEMU(II) = ENNEMU(II) + $ XVAL(IALPHA)*SCALE_SIGMA*MU_AI(II,IALPHA)
      ENNEWU(II) = ENNEWU(II) + $ SI_AI(II,IALPHA)*SI_AI(II,IALPHA) + $ MU_AI(II,IALPHA)*MU_AI(II,IALPHA)* $ XVAL(IALPHA)*SCALE_SIGMA
   END DO
   RADICE(II) = ENNEWU(II)/ENNEMU(II)/ENNEMU(II) - 1.D0/ENNE_TOT
   IF(UN_NALPHA_NEGATIVO) THEN
      FVEC(II) = 1.D2*ABS(RADICE(II))
   ELSEIF(RADICE(II).LT.0) THEN
      WRITE(*,*) 'II= ',II, ' RAD= ',RADICE(II), $ ' ENNEMU= ',ENNEMU(II), ' ENNEWU= ',ENNEWU(II)
      FVEC(II) = 1.D6*ABS(RADICE(II))
   ELSE
      RADICE(II) = SQRT(RADICE(II))
      FVEC(II) = (RADICE(II)-SIGMA_REL_VAR)/0.1/SIGMA_REL_VAR
   END IF
   END DO
END DO

C also jacobian calculation
C
IF( IFLAG.EQ.2) THEN
   DO II=1,NVOX_CTV
      DO IALPHA=1,NPENOK
         AUS1 = (MU_AI(II,IALPHA)*MU_AI(II,IALPHA) + $ SI_AI(II,IALPHA)*SI_AI(II,IALPHA))/ $ ENNEMU(II)/ENNEMU(II)
         AUS2 = ENNEWU(II)/ENNEMU(II)*ENNEMU(II)*ENNEMU(II)* $ MU_AI(II,IALPHA)
         IF(RADICE(II).LT.0) THEN
            FJAC(II,IALPHA) = 1.D2*ABS(RADICE(II))
         ELSEIF(NALPHA_NEGATIVO(IALPHA)) THEN
            FJAC(II,IALPHA) = XVAL(IALPHA)*XVAL(IALPHA)
         ELSE
            FJAC(II,IALPHA) = (AUS1 -2*AUS2 + $ 1./ENNE_TOT/ENNE_TOT)/0.1/SIGMA_REL_VAR/2./RADICE(II)
         END IF
      END DO
   END DO
   END DO
END IF
RETURN
END
implicit none
include 'fluence.inc'

c double precision futil
EXTERNAL FCNMINUIT, futil
double precision stval(NPENMAX), step(NPENMAX)
double precision ARGLIS(NPENMAX)
character*10 parname
integer ii, ierflg, istat, npari, nparx, IVARBL
double precision chi2, fedm, errdef, val
double precision ERROR, BND1, BND2
c write(6, 888)
CALL MNINIT (5, 6, 7)
CALL MNSETI('FLUKA-TPS flux minimization')
c
parname = 'p000'
do ii = 1, npenok
   stval(ii) = flux0(ii)/nsciami
   step(ii) = stval(ii)/100
   if(ii.lt.npenok) then
      WRITE(parname(4:4),'(I1)')ii
   elseif(ii.lt.100.) then
      WRITE(parname(3:4),'(I2)')ii
   elseif(ii.lt.1000.) then
      WRITE(parname(2:4),'(I3)')ii
   endif

   CALL MNPARM(ii, parname, STVAL(ii), STEP(ii), 0., 1.d5, IERFLG)
end do

c arglis(1) = -1.
CALL MNEXCM(FCNMINUIT, 'SETPRINTOUT', ARGLIS, 1, IERFLG, futil)
arglis(1) = 2.
CALL MNEXCM(FCNMINUIT, 'SETSTRATEGY', ARGLIS, 1, IERFLG, futil)
arglis(1) = 1.0D5
CALL MNEXCM(FCNMINUIT, 'SIM', ARGLIS, 1, IERFLG, futil)
CALL MNEXCM(FCNMINUIT, 'IMPROVE', ARGLIS, 0, IERFLG, futil)
CALL MNEXCM(FCNMINUIT, 'MINI', ARGLIS, 0, IERFLG, futil)
CALL MNSTAT(chi2, FEDM, ERRDEF, NPARI, NPARX, ISTAT)
write(6, *) 'chi2= ', chi2, ' ,FEDM= ', fedm, ' errdef= ', ERRDEF,
$ ' npari= ', NPARI, ' nparx= ', NPARX, ' istat= ', ISTAT
c
do ii = 1, npenok
   CALL MNPOUT (ii, parname, VAL, ERROR, BND1, BND2, IVARBL)
   flusso(ii) = val*nsciami
   write(6, *) 'par # ', ii, ' flusso= ', flusso(ii)
end do

c 888 format(/,'minimization using minuit',/)
c return
end

c-------------------------------------------------------------
SUBROUTINE FCNMINUIT(NPAR, GRAD, FVAL, XVAL, IFLAG, FUTIL)
c-------------------------------------------------------------
implicit none
integer Npar, iflag, ialpha
double precision fval, futil, xval_old, fun_old, gradiente
DOUBLE PRECISION grad(*), xval(*)
double precision epsi
external futil, gradiente


parameter (epsi=1.d-6)
c
read input data, calculate any necessary constants, etc.

IF (IFLAG .EQ. 1) THEN
ENDIF

c
calculate GRAD, the first derivatives of FVAL (this is optional)

IF (IFLAG .EQ. 2) THEN
write(6,765)
  fun_old = futil(npar,xval)
  do ialpha = 1,NPAR
      xval_old = xval(ialpha)
      xval(ialpha) = xval(ialpha)*(1.0d0+epsi)
      grad(ialpha) = ( futil(npar,xval) - fun_old )/epsi
      xval(ialpha) = xval_old
  end do
write(6,654)ialpha,grad(ialpha),gradiente(ialpha,xval)
ENDIF

c
Fval = futil(npar,xval)

will come here only after the fit is finished.
C Perform any final calculations, output fitted data, etc.

If (IFLAG .EQ. 3) THEN
ENDIF

c
765 format('calcolo del gradiente')
654 format('comp #',1x,i3,2x,'numerico=',2x,d8.4,2x,'analitico',
          $     1x,d8.4)

c
RETURN
END

c---------------------------------------------------
double precision function FUTIL(npar,xval)
c---------------------------------------------------
C It is responsability of user to pass any parameter values needed
c either through arguments, or in a COMMON block
c implicit none
include 'fluence.inc'
icntger npar, ii, ialpha
double precision xval(*)
double precision chi2, somma, delta, sig_delta
c
chi2 = 0.0d0
c
do ii = 1,nvox_CTV
  somma = 0.d0
  do ialpha = 1, npar
      somma = somma + xval(ialpha)*MU_AI(ii,ialpha)*nsciami
  end do
  delta = dose0(ii) - somma
  sig_delta = dose0(ii)*sigma_perc_flux
  chi2 = chi2 + delta*delta/sig_delta/sig_delta
end do
futil = chi2
SUBROUTINE FCNSIG_MINUIT(NPAR, GRAD, FVAL, XVAL, IFLAG, FUTIL_SIG)

implicit none
integer Npar, iflag, ialpha
double precision fval, futil_sig, xval_old, fun_old
DOUBLE PRECISION grad(*), xval(*)
double precision epsi
external futil_sig

return
end

c------------------------------------------------------------------
c subroutine mini_sigma_minuit()
c------------------------------------------------------------------
c minimizzazione della sigma tramite MINUIT in Fortran callable mode
c implicit none
include 'fluence.inc'
double precision futil_sig
EXTERNAL FCNSIG_MINUIT, futil_sig
double precision stval(NPENMAX), step(NPENMAX)
double precision ARGLIS(10)
character*10 parname
integer ii, ierflg, istat, npari, npax, IVARBL
double precision chi2, fedm, errdef, val
double precision ERROR, BND1, BND2
c
CALL MNINIT (5, 6, 7)
CALL MNSETI('FLUKA-TPS sigma minimization')
c
parname = 'p000'
do ii = 1, npenok
   stval(ii) = flux_sig(ii)/nsciami
   step(ii) = stval(ii)/100
   if(ii.lt.10) then
      WRITE(parname(4:4),'(I1)')ii
   elseif(ii.lt.100.) then
      WRITE(parname(3:4),'(I2)')ii
   endif
   CALL MNPARM (ii, parname, STVAL(ii), STEP(ii), 0., 1.d4, IERFLG)
end do
c
arglis(1) = 2.
CALL MNEXCM(FCNSIG_MINUIT, 'SETSTRATEGY', ARGLIS, 1, IERFLG, futil_sig)
arglis(1) = 1.0D4
CALL MNEXCM(FCNSIG_MINUIT, 'SIM', ARGLIS, 1, IERFLG, futil_sig)
c
CALL MNSTAT(chi2, FEDM, ERRDEF, NPARI, NPAX, ISTAT)
write(6,*)'chi2= ',chi2,' ,FEDM= ',fedm,' errdef= ',ERRDEF,
$     ' npari= ',NPARI,' npax= ',NPAX,' istat= ',ISTAT
c
do ii = 1, npenok
   CALL MNPOUT (ii, parname, VAL, ERROR, BND1, BND2, IVARBL)
   flux_sig(ii) = sngl(val)*nsciami
   write(6,*)'flux_sig= ',flux_sig(ii)
end do
c
return
end

SUBROUTINE FCNSIG_MINUIT(NPAR, GRAD, FVAL, XVAL, IFLAG, FUTIL_SIG)

implicit none
integer Npar, iflag, ialpha
double precision fval, futil_sig, xval_old, fun_old
DOUBLE PRECISION grad(*), xval(*)
double precision epsi
external futil_sig

return
end
parameter (eps=1.d-6)

c read input data, calculate any necessary constants, etc.
c
IF (IFLAG .EQ. 1) THEN
ENDIF

c calculate GRAD, the first derivatives of FVAL (this is optional)
c
IF (IFLAG .EQ. 2) THEN
  fun_old = futil_sig(npar,xval)
  do ialpha = 1,NPAR
    xval_old = xval(ialpha)
    xval(ialpha) = xval(ialpha)*(1.0d0+eps)
    grad(ialpha) = ( futil_sig(npar,xval) - fun_old )/eps
    xval(ialpha) = xval_old
  end do
ENDIF

c Always calculate the value of the function, FVAL, which is usually
C a chisquare or log likelihood. Optionally, calculation of FVAL may involve
C
Fval = futil_sig(npar,xval)
C
will come here only after the fit is finished.
C Perform any final calculations, output fitted data, etc.
c
IF (IFLAG .EQ. 3) THEN
ENDIF

c RETURN
END

c-----------------------------------------------------------
double precision function FUTIL_SIG(npar,xval)
c-----------------------------------------------------------
C It is responsibility of user to pass any parameter values needed
C either through arguments, or in a COMMON block

c implicit none
include 'fluence.inc'
integer npar, ii, ialpha
DOUBLE PRECISION xval(*)
double precision chi2, enne_tot, elmat, aus
double precision amedia(NBINMAX), sigmed2(NBINMAX)
double precision delta, sig_delta
c
call vzero(amedia,2*NBINMAX)
call vzero(sigmed2,2*NBINMAX)
c
chi2 = 0.000
enne_tot = 0
do ialpha=1,npar
  enne_tot = enne_tot + xval(ialpha)*nsciami
end do

do ii=1,nvox_CTV
do ialpha=1,npar
  elmat = si_ai(ii,ialpha)*si_ai(ii,ialpha) +
           mu_ai(ii,ialpha)*mu_ai(ii,ialpha)
  sigmed2(ii) = sigmed2(ii) + elmat*xval(ialpha)*nsciami
end do
sigmed2(ii) = sigmed2(ii)/enne_tot

c aus = 0
do ialpha = 1,npenok
   aus = aus + xval(ialpha)*mu_ai(ii,ialpha)*nsci_am
end do
amedia(ii) = aus/enne_tot
sigmed2(ii) = (sigmed2(ii) - amedia(ii)*amedia(ii))/enne_tot

c delta = dsqrt(sigmed2(ii))/amedia(ii)-sigma_rel_var
sig_delta = 0.2*sigma_rel_var

c chi2 = chi2 + delta*delta/sig_delta/sig_delta
end do
futil_sig = chi2
return
end