Dissertation

Submitted to the
Combined Faculties for the Natural Sciences and for Mathematics
of the Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

presented by

Edgardo Dörner

Born in Santiago, Chile

Oral examination: 20 July, 2012

Development and Validation of a Multi-Leaf Collimator Component Module Using a Modified EGSnrc Platform for Monte Carlo Simulations

REFEREES:

Prof. Dr. Günther Hartmann

Prof. Dr. Wolfgang Schlegel

To Pili…

Zusammenfassung

Für den Monte Carlo Code BEAMnrc wurde ein neues “Component Module“ genannt MMLC entwickelt. Es modelliert die Geometrie und die dosimetrischen Eigenschaften des Miniatur-Multileaf-Kollimators ModuLeaf (MRC Systems GmbH, Heidelberg). Das neues Component Module basiert zum Teil auf einem bereits existierenden “Component Module“ genannt DYNVMLC, das jedoch wegen der besonderen Bauweise des vorliegenden Miniatur-Kollimators für die Simulierung nicht geeignet war. Die Entwicklung wurde auf einer modifizierten Monte Carlo Plattform durchgeführt, die auf der EGSnrc Plattform des National Research Council of Canada beruht. Die neue Plattform ermöglichte es, in der Microsoft Visual Studio Umgebung zu arbeiten und hierbei den modernen Visual Fortran Compiler von Intel einzusetzen.

Die Modellierung der dosimetrischen Eigenschaften des Miniatur-Multileaf-Kollimators ModuLeaf wurde in Verbindung mit der 6 MV Röntgenstrahlung des Linearbeschleunigers PRIMUS durchgeführt, um nachprüfbare dosimetrische Ergebnisse zu erzielen. Die Validierung der Ergebnisse erfolgte durch zwei Verfahren: (a) durch ein Ray-Tracing Verfahren, um die korrekte Geometrie des MLC zu überprüfen, und (b) durch den Vergleich der berechneten und mit gemessenen Ergebnisse der folgenden dosimetrischen Parameter: Output-Faktoren, Dosisprofile, isozentrische Position des Feldrands, Größe des Halbschattens, Interleaf-Leckage und Transmission. Eine ausgezeichnete Übereinstimmung wurde für alle Parameter gefunden. Es wurde insbesondere auch gefunden, dass die Beziehung zwischen der mechanischen Positionierung der Kollimatorlamellen und des isozentrischen Strahlenfeldrandes, der von der Form der Lamellen-Enden abhängt, mit einer höheren Genauigkeit im Vergleich zur messtechnischen Methode untersucht werden kann. Dies demonstriert die weitere Nützlichkeit des neuen Monte Carlo Moduls.

Abstract

A new component module (CM) named MMLC was developed for the Monte Carlo code BEAMnrc. It models the geometry and the dosimetrical characteristics of the add-on miniature multileaf collimator ModuLeaf (MRC Systems GmbH, Heidelberg, Germany, now part of Siemens, Erlangen, Germany). The new component module is partly based on the existing CM called DYNVMLC. The development was performed using a modified EGSnrc platform which enables to work in the Microsoft Visual Studio environment.

In order to validate the new CM, the PRIMUS linac with 6 MV x-rays (Siemens OCS, Concord, California, USA) equipped with the ModuLeaf mini MLC was modeled. Validation was performed by two methods: (a) a ray-tracing method to check the correct geometry of the MLC, and (b) a comparison of calculated and measured results of the following dosimetrical parameters: output factors, dose profiles, field edge position, penumbra, MLC interleaf leakage and transmission values. An excellent agreement was found for all parameters. It was in particular also found that the relationship between leaf position and field edge depending on the shape of leaf ends can be investigated with a higher accuracy by this new calculation tool than by measurements further demonstrating the usefulness of the new CM.

CONTENT

1 INTRODUCTION

This work describes the development of a calculation model for a multi-leaf collimator which represents a key component of modern radiotherapy units using a modified EGSnrc Monte Carlo platform. The original EGSnrcMP is a multi-platform offered by the National Research Council of Canada for running the EGSnrc Monte Carlo code system. Its purpose is the simulation of the transport and energy deposition of photons and electrons and hereby allowing the investigation of various problems in radiotherapy and dosimetry (Kawrakow and Rogers 2003, Kawrakow *et al* 2004, Rogers 2006). The EGSnrc code system is copyrighted material owned by the National Research Council of Canada. Being offered by the NRC website for free implementation anywhere and successfully applied as demonstrated in many publications, it has obtained a great reputation over the years (Rogers 2006). One of the reasons for its success was that great efforts have been put into implementation tools to run the Monte Carlo codes under various conditions of compilers and/or operating systems (Kawrakow *et al* 2004).

However, although the web-based implementation of this EGSnrcMP platform is well organized, it has still three disadvantages which makes the start of Monte Carlo work with this system cumbersome:

- The various MC application codes are more or less rigidly fixed. Individual code changes are not easily to perform. In consequence, they are frequently used like "black box" tools for already well defined categories of calculation tasks.
- Written in MORTRAN and FORTRAN, the compilation of executable programs is optimized using the public GNU FORTRAN Compiler. This is certainly perfect for zero budget projects. However, nowadays there are also quite advanced alternative compilers commercially available such as the Intel Visual FORTRAN compiler (IVFC) for WINDOWS. The option to adapt EGSnrcMP to such modern compilers is currently less convenient. On the other hand, a strong motivation to work with the IVFC would be that this compiler allows a much more transparent code development in FORTRAN.
- The IVFC can be embedded in Visual Studio Platform of Windows. Thus its debugging capacity for code and running error analysis can be utilized which further greatly facilitates code changes and developments according to own ideas.

Therefore, a modified environment was developed as a Monte Carlo working tool which enables an easier implementation of the EGSnrc Monte Carlo code system to run with the IVFC under Windows. In the following, this particular MC environment is referred to as "DKFZ platform".

The DKFZ platform was then used to develop a new component module (CM) for a particular mini multileaf collimator (mini MLC) to be included in the BEAMnrc code. The BEAMnrc code is a special Monte Carlo program to simulate the head of a clinical electron accelerator including the target for Bremsstrahlung production and the collimator system (primary collimator, dose monitor, secondary rectangular collimator jaws or a MLC system). The particular components which are relevant to produce the electron and/or photon beam of a linear accelerator are dealt with by so-called component modules. In the past, MLCs that are integrated in the treatment head often have a leaf width in the order of 1 cm at the isocenter distance of 100 cm. In certain cases a better resolution was desirable, as in the case of small treatment volumes or lesions near organs at risk (OAR). Based on theoretical considerations it was found that a leaf width around 2-3 mm would be the optimum choice for a MLC (Bortfeld et al 2000). In accordance to these findings, the mini MLC called ModuLeaf has

been developed as an add-on MLC to the accelerator head providing those characteristics (MRC Systems GmbH, Heidelberg, Germany, now part of Siemens, Erlangen, Germany). For this particular mini MLC a component module was not available so far.

The new CM was developed and validated for the BEAMnrc user code, to model the dosimetric characteristics of the ModuLeaf mini MLC. The new component module was partly based on an existing component module called DYNVMLC CM. It was originally developed to model the Varian Millennium 120 leaf collimator (Heath and Seuntjens 2003). However, it was found that this existing CM was not appropriate to model the Moduleaf mMLC. The new CM for the ModuLeaf was finally validated by comparing different dosimetrical parameters obtained with this new CM with those obtained by measurements and published as a dosimetric characterization of the ModuLeaf mini MLC (Hartmann and Föhlisch 2002).

The motivation for this work was:

- To demonstrate the great improvements and facilitation for entering the field of Monte Carlo calculations in Medical physics when using the DKFZ platform on hand of the challenging development of a new compound module.
- To investigate how well the dosimetric properties of the ModuLeaf mini MLC can be reproduced by Monte Carlo simulations.
- To demonstrate the usefulness and even the superiority of calculated dosimetrical characteristics of the ModuLeaf mini MLC against measured results.

2 METHODS AND MATERIALS

This section begins with an overview of the physical background needed to describe the energy transport and absorption of ionizing radiation. Then the Monte Carlo technique used for the simulation of the electromagnetic radiation is presented, in particular the DKFZ Monte Carlo platform. Finally, the characteristics of the ModuLeaf component module developed for the Monte Carlo platform are detailed, and the validation procedures carried out are summarized.

2.1 The physics of ionizing radiation

Radiation dosimetry deals with methods for a quantitative determination of energy deposited in a given medium by directly or indirectly ionizing radiations. Although the physics of ionizing radiation is well known and well described in various textbooks such as “Radiation Oncology Physics: A Handbook for Teachers and Students” (Podgorsak 2005), “Handbook of Radiotherapy Physics: theory and practice” (Mayles et al 2007) and “Introduction to Radiological Physics and Radiation Dosimetry” (Attix 1986), a short overview on radiation field quantities and the interaction of ionizing radiation with matter is given below for completeness. This is considered as particular helpful for a thorough understanding of the tracking and energy scoring procedures within the Monte Carlo code used in this work. Whenever text pieces have been adopted from a particular textbook, it is made traceable by giving the reference to this textbook.

High energy particles impart their energy to matter through different interactions processes with the medium atoms. In these processes the direction and energy of the particles change and even a total absorption of a particle or the production of a new particle may occur. These interactions are stocastical in nature and are described by a probability distribution such as differential and total cross sections together with stopping powers. All these concepts must be taken into account in MC simulations and are described in this section.

2.1.1 Important quantities

2.1.1.1 Radiation field quantities: particle and energy fluence

Radiation measurements and investigations of radiation effects require various specifications of the radiation field at the point of interest. Particle fluence is a very important basic quantity, involving the number of incident particles per unit area. The concept is illustrated in Figure 2.1.

N is the number of particles striking a finite sphere surrounding P during a finite time interval. If the sphere is reduced to an infinitesimal one at P with a cross sectional area of dA, then the fluence Φ is given by

$$\Phi = \frac{dN}{dA} \tag{1}$$

The unit of particle fluence is m^{-2} (ICRU 1998). The use of a sphere of cross-sectional area $\mathrm{d}A$ expresses in the simplest manner the fact that one considers an area $\mathrm{d}A$ perpendicular to the direction of each particle and hence that particle fluence is independent of the incident angle of the radiation (Podgorsak 2005).

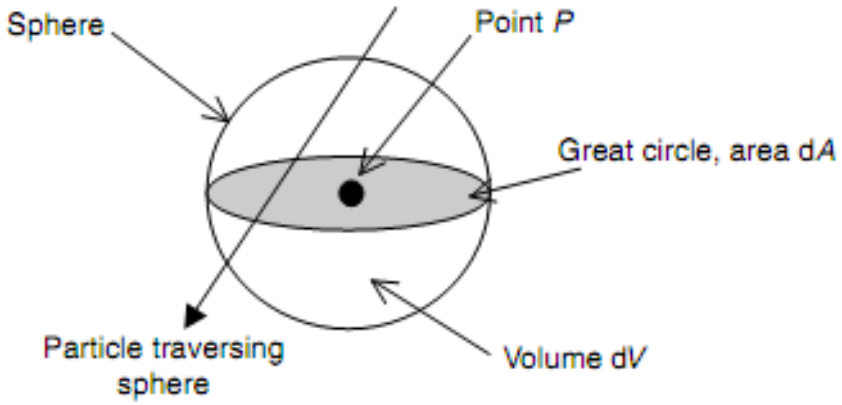

Figure 2.1: Particle fluence definition. Characterization of the radiation field at a point P in terms of the radiation traversing a sphere centred at P (Attix 1986).

The energy fluence Ψ is the quotient of dE by dA, where dE is the radiant energy incident on a sphere of cross-sectional area dA:

$$\Psi = \frac{dE}{dA} \tag{2}$$

The unit of energy fluence is J/m^2 (ICRU 1998). Energy fluence can be calculated from particle fluence by using the following relation:

$$\Psi = \frac{dN}{dA}E = \Phi E \tag{3}$$

where E is the energy of the particle and dN represents the number of particles with energy E. Almost all realistic particle beams are polyenergetic, hence the above defined concept need to be applied to such beams. The concepts of particle fluence spectrum (i.e. particle fluence differential in energy) and energy fluence spectrum (i.e. energy fluence differential in energy) are introduced. They are defined respectively as:

$$\Phi_{\mathrm{E}}(E) \equiv \frac{d\Phi}{dE}(E) \tag{4}$$

and

$$\Psi_{\mathrm{E}}(E) \equiv \frac{d\Psi}{dE}(E) = \frac{d\Phi}{dE}(E)E \tag{5}$$

As the particle beams are not necessarily isotropic the concept of particle fluence differential in solid angle is introduced. It is defined as:

$$\Phi_{\vec{\Omega}}(\vec{\Omega}) \equiv \frac{d\Phi}{d\vec{\Omega}}(\vec{\Omega}) \tag{6}$$

where $\vec{\Omega}$ is the directional vector of the solid angle Ω. The fluence differential in solid angle multiplied with $\vec{\Omega}$ results in the vectorial fluence differential in solid angle:

$$\vec{\Phi}_{\vec{\Omega}}(\vec{\Omega}) = \vec{\Omega}\,\Phi_{\vec{\Omega}}(\vec{\Omega}) \tag{7}$$

The particle fluence can also be expressed as the quotient of the track-length density (i.e. track-length per volume) of particles at a point in space within a small volume dV (Chilton 1978):

$$\Phi = \frac{dL(\vec{r})}{dV} \tag{8}$$

The prove that this definition is equivalent to Equation (1) is given in Appendix A. The expression of the particle fluence as the quotient of the track-length density is extremely useful for MC simulations. Using Equation (8) the particle fluence can be derived by "book-keeping" the track lengths of the particles going through a region of interest.

2.1.1.2 Interaction quantities

2.1.1.2.1 Total cross sections

Particles interact with various target entities such as atomic electrons, nuclei, atoms or molecules. The probability of interaction is usually expressed in terms of the cross section σ, that represents the cross sectional area that the target presents in the plane normal to the incident photon direction and is used to estimate the probability of interaction (Mayles et al 2007).

The total interaction cross section, independent of which process occurs, is the sum of the cross sections for the individual processes. For example, in the case of photons

$$\sigma = \sigma_{pe} + \sigma_{coh} + \sigma_{incoh} + \sigma_{pair} \tag{9}$$

where σ_{pe}, σ_{coh}, σ_{incoh} and σ_{pair} are the interaction cross sections for photo-electric absorption, coherent and incoherent scattering, and pair production, respectively. These effects are detailed in Section 2.1.2. The target entity may be an atomic electron, a nucleus, an atom or a molecule. Therefore care must be taken to distinguish between these targets. For example, the relation between the cross-section per electron ${}_e\sigma$ and per atom ${}_a\sigma$ is given by ${}_a\sigma = Z \times {}_e\sigma$ where Z is the atomic number of the atom (Mayles et al 2007). The unit of cross section is cm^2, although the barn is still frequently used (1 barn = 10^{-24} cm^2 = 10^{-28} m^2).

Photons incident on an absorber will either interact in it or else pass through it without interacting. The number of photons transmitted undisturbed through an absorber of thickness t of a given element and density is equal to (see Figure 2.2):

$$\Phi(t) = \Phi_0 \mathrm{e}^{-\mu t} \tag{10}$$

The linear attenuation coefficient μ is a property of the material and depends on photon energy. It is the probability per unit length for interaction and is related to the total cross-section σ according to

$$\mu = N\sigma \tag{11}$$

where N is the number of target entities per unit volume of the absorber. It is given by $(N_A/A) \times \rho$, where N_A is the Avogadro's number, A the relative atomic mass of the absorber element and ρ its density.

The penetration power of a photon beam in an absorber is commonly expressed by means of the mean free path. This is defined as the average distance $\bar{x}$ travelled by the photon before it interacts. For mono-energetic photons it is given by:

$$\bar{x} = \int_0^\infty x e^{-\mu x} dx = \frac{1}{\mu} \tag{12}$$

The mean free path equals the thickness of an absorber that reduces the primary photons to a fraction $1/e = 37\%$ of their initial number.

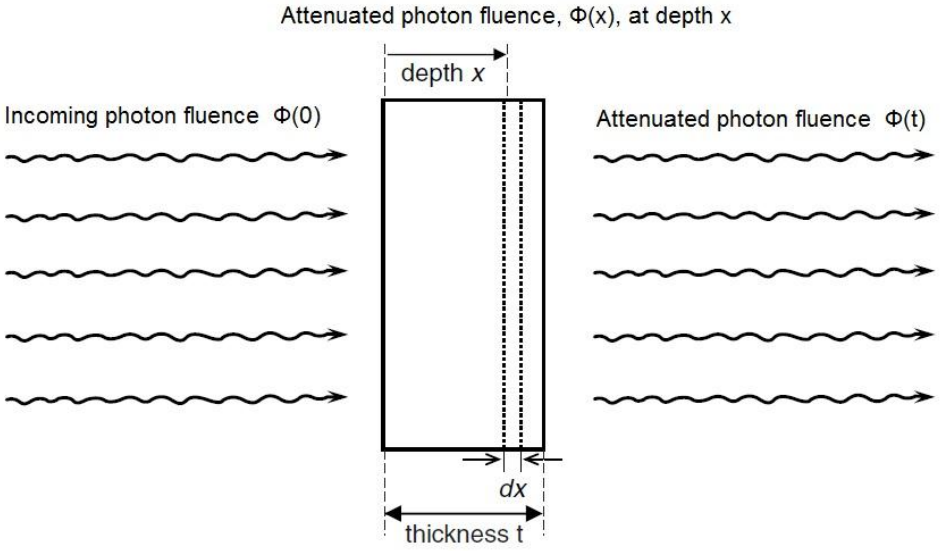

Figure 2.2: Calculation of photon attenuation through an absorber. The incident fluence of primary photons $\Phi_0 = \Phi(0)$ is reduced to a fluence $\Phi(x)$ at depth x in the absorber (Mayles et al 2007).

2.1.1.2.2 Differential cross sections

The distribution of scattered particles during an interaction process may not be isotropic, with their distribution related in some fashion to the direction of the incoming photon and its energy. In order to quantify such effects the cross section is related to the solid angle $\vec{\Omega}$ in the direction of the scattered photon, introducing the concept of differential cross section $d\sigma/d\vec{\Omega}$. Therefore the probability that the photon scatters into a solid angle $d\vec{\Omega}$ is $\left(d\sigma/d\vec{\Omega}\right)d\vec{\Omega}$ and it follows that (Mayles et al 2007):

$$\sigma = \int_{4\pi} \frac{d\sigma(\theta,\phi)}{d\vec{\Omega}} d\vec{\Omega} \tag{13}$$

where θ is the scattering polar angle and ϕ is the azimuthal angle. This concept can be further generalized. For example, cross sections differential in the energy of the scattered photon, $d\sigma/d(h\nu')$, or cross sections differential in both energy and angle, $d\sigma/d\vec{\Omega}d(h\nu')$ can also be of interest.

2.1.1.2.3 Stopping power

An important element used to describe the charged particle behaviour in a medium is the stopping power. The linear stopping power $S(x,E)$ is defined as the expectation value of the rate of energy loss per unit path length (dE/dx) of the charged particle. The mass stopping power is defined as the linear stopping power divided by the density of the absorbing

medium (Podgorsak 2005). Typical units for the linear and mass stopping powers are MeV/cm and MeVcm^2/g, respectively. Stopping powers are widely used in radiation dosimetry (e.g. for dose calculation), but they are rarely measured and must be calculated from theory. For electrons and positrons the Bethe theory is used to calculate stopping powers (Podgorsak 2005).

Total stopping power is a sum of collision stopping power S_c and radiative stopping power S_r. The collision stopping power is the average energy loss per unit path length due to inelastic collisions while the radiative stopping power is the average energy loss per unit path length due to the emission of photons in a bremsstrahlung process.

2.1.1.3 Dosimetrical quantities

The energy absorbed from the ionizing radiation is usually described by the absorbed dose D, which is the mean energy $\bar{\varepsilon}$ imparted by ionizing radiation to an unit of mass m in a finite volume V, i.e. (Podgorsak 2005)

$$D = \frac{d\bar{\varepsilon}}{dm} \tag{14}$$

The energy imparted $\bar{\varepsilon}$ is the sum of all the energy entering the volume of interest minus all the energy leaving it, taking into account any mass-energy conversion within the volume. The unit of absorbed dose is joule per kilogram (J/kg). The name for the unit of absorbed dose is the gray (Gy).

Under the conditions that (a) radiative photons escape the volume of interest and (b) secondary electrons are absorbed on the spot (or there is a charged particle equilibrium (CPE) of secondary electrons), the absorbed dose to the medium D_{med} is related to the electron fluence Φ in the medium as follows (Podgorsak 2005):

$$D_{med} = \Phi\left(\frac{\mathrm{S_c}}{\rho}\right) \tag{15}$$

where S_c/ρ is the mass collision stopping power of the medium at the energy of the electron. Due to electron slow-down in a medium, even for a monoenergetic starting electron kinetic energy E_K there is always a primary fluence spectrum present that ranges in energy from E_K down to zero. In this case the absorbed dose to the medium can be obtained by:

$$D_{med} = \int_0^{E_K} \Phi_{\mathrm{E}}(\mathrm{E})\left(\frac{\mathrm{S_c}}{\rho}\right)(E)dE \tag{16}$$

2.1.2 Interaction of photons with matter

The description of the details of interactions of ionizing radiation with matter is essentially following the “Handbook of Radiotherapy Physics: theory and practice” from Mayles, Nahum, and Rosenwald (Mayles et al 2007) and the PIRS Report 701: “The EGSnrc Code System: Monte Carlo Simulation of Electron and Photon Transport” (Kawrakow and Rogers 2003). Both descriptions are in particular helpful to be directly applied into the various Monte Carlo routines handling the interaction processes.

Photons interact with surrounding matter via four basic processes: materialisation into an electron/positron pair in the electromagnetic field of the nuclei and surrounding atomic electrons, incoherent (Compton) scattering with atomic electrons, photo-electric absorption and coherent (Rayleigh) scattering with the molecules (or atoms) of the medium. The cross

sections in water for these processes are shown in Figure 2.3. The first three interaction processes transfer energy from the photon radiation field to electrons, one of them dominates depending on energy and the medium in which the transport takes place. The pair production process dominates at high energies. At some intermediate energies incoherent scattering is the most important process, at low energies the photo-electric process dominates (Kawrakow and Rogers 2003).

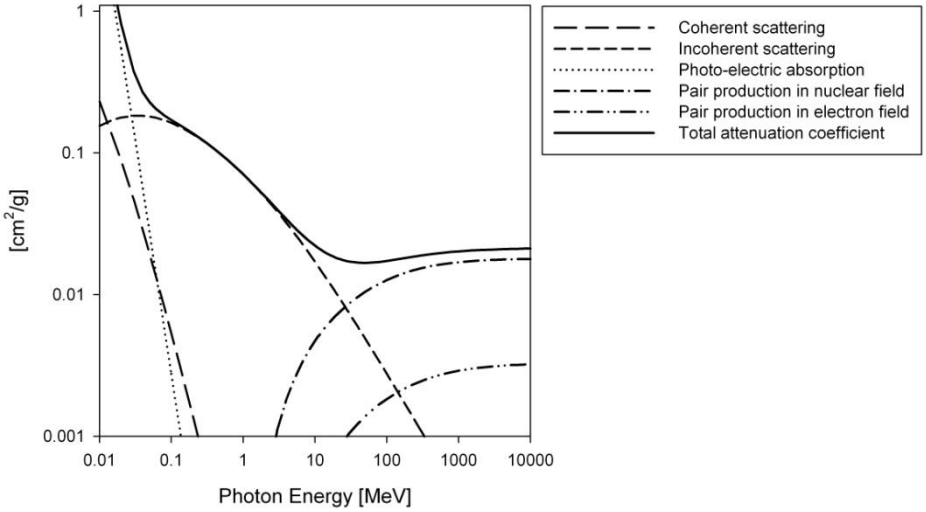

Figure 2.3: Photon mass attenuation coefficients for water (XCOM NIST data).

Photon interactions can be characterized as absorption or scattering processes. In a full absorption process, the incoming photon loses all its energy and the energy is transferred to the target entity. Secondary particles are emitted during or subsequently to the interaction. In a full scattering process, an incoming photon interacts with a target entity and its direction of motion, energy and momentum may be changed because of this interaction. The main absorption processes are photoelectric absorption and pair production. The main scattering processes are coherent and incoherent scattering (Mayles et al 2007).

Figure 2.4 shows that the mean-free path for photons with energies between 10 keV and 40 MeV are of the order of 20 cm for water. The photons will interact only a few times in macroscopic objects such as a radiation measuring device, a tank of water or a patient undergoing treatment with radiotherapy. If one considers the transport of photons alone, it is feasible, with modest computational resources, to simulate hundreds of millions of particle histories.

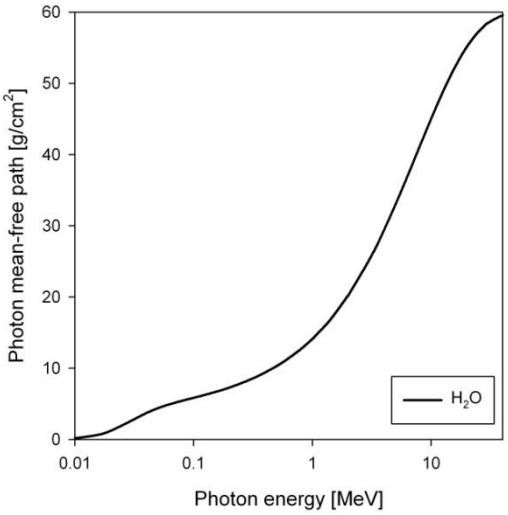

Figure 2.4: Mean free-path of photons in water (XCOM NIST data).

2.1.2.1 Photo-electric absorption

In the photo-electric absorption process a photon is absorbed by an atom and an electron is emitted with an energy given by the incident photon energy minus its binding energy. The atom, left in an excited state with a vacancy in the ionised shell, relaxes via the emission of fluorescent (characteristic) photons and/or Auger electrons (Kawrakow and Rogers 2003). A diagram of the process is illustrated in Figure 2.5.

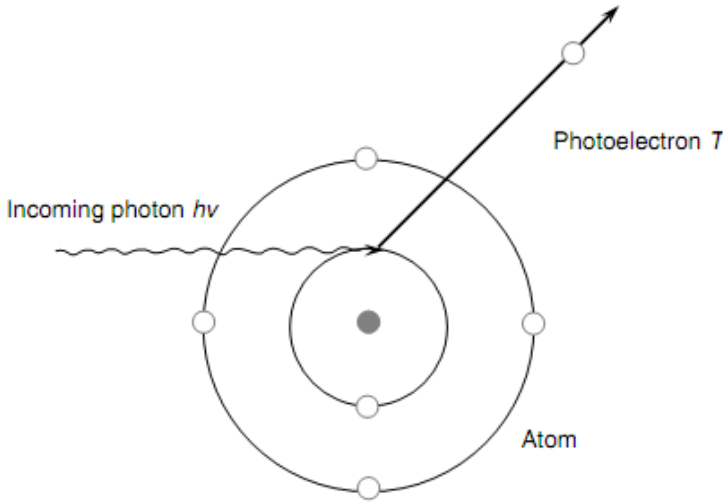

Figure 2.5: Photo-electric absorption. An incoming photon with energy $h\nu$ interacts with the atom and ejects a photoelectron with kinetic energy T (Mayles et al 2007).

In general, the cross-section σ_{pe} for photo-electric absorption increases strongly with decreasing photon energy. Figure 2.6 shows this cross-section for lead. The cross-section displays a series of discontinuities at energies corresponding to the binding energies of the electrons in the atomic shells. These discontinuities are known as absorption edges. At energies above the K absorption edge about 80% of the interactions take place in the K shell (Mayles et al 2007).

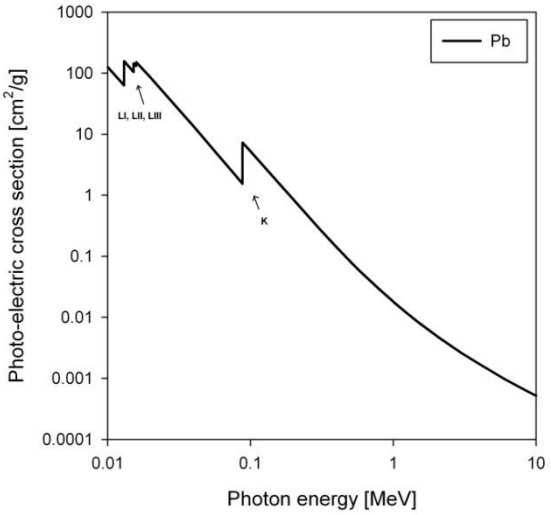

Figure 2.6: The total photo-electric absorption cross-section for lead as a function of photon energy. The positions of the L and K absorption edges are indicated. (XCOM NIST data).

The cross-section for photo-electric absorption depends strongly on atomic number. Above the K absorption edge, the cross-section per atom as a function of photon energy and atomic number is approximately given by (Mayles et al 2007)

$$_a\sigma_{pe} \cong kZ^4/(h\nu)^3 \tag{17}$$

which explains the strong impact of this process at low photon energies and particularly at high atomic numbers.

2.1.2.2 Incoherent (Compton) scattering

In incoherent scattering the photon transfers part of its energy to an atomic electron that is ejected from the atomic shell. The incoming photon with energy $h\nu$ is scattered through the scattering angle θ. The process was first described by Compton who assumed the electron to be free and at rest at the moment of collision. In this approximation the process is also known as Compton scattering. The energy $h\nu'$ of the scattered photon is given by the Compton equation:

$$h\nu' = \frac{h\nu}{1 + \alpha(1 - \cos\theta)} \tag{18}$$

where $\alpha = h\nu/m_0c^2$ and m_0 is the rest mass of the electron. The incoherent scattering process is depicted in Figure 2.7.

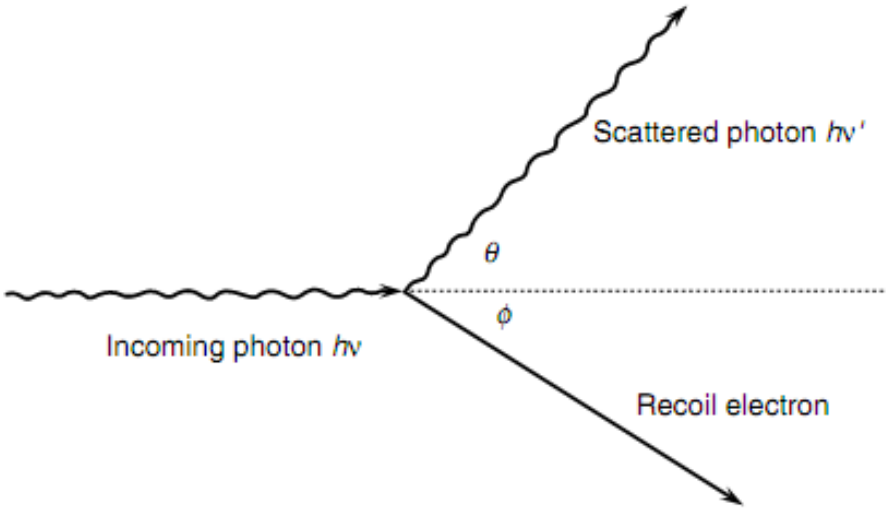

Figure 2.7: Scattering angle and energies for incoherent scattering. An incoming photon with energy $h\nu$ scatters to produce a scattered photon of energy $h\nu'$ and a recoil electron (Mayles et al 2007).

The photon loses no energy when it is scattered in the forward direction ($\theta = 0°$). The maximum loss of energy occurs when the photon is backscattered ($\theta = 180°$). The recoil electron is expelled with a kinetic energy $T_e = h\nu - h\nu'$. This energy is zero when the photon scatters in the forward direction and takes its largest value, $T_e = h\nu \cdot 2\alpha/(1 + 2\alpha)$, when the photon is backscattered. Figure 2.8 shows the maximum kinetic energy of the electron expressed as a fraction of the incident photon energy. At low photon energies, the energy transfer is small and at high energies the energy of the photon is almost totally transferred to the electron.

The angle ϕ between the directions of the incoming photon and the electron is given by (Mayles et al 2007):

$$\cot\phi = (1 + \alpha)\tan\frac{\theta}{2} \qquad (19)$$

This equation shows that $\phi \leq \pi/2$ and the recoil electron is never emitted in the backward direction.

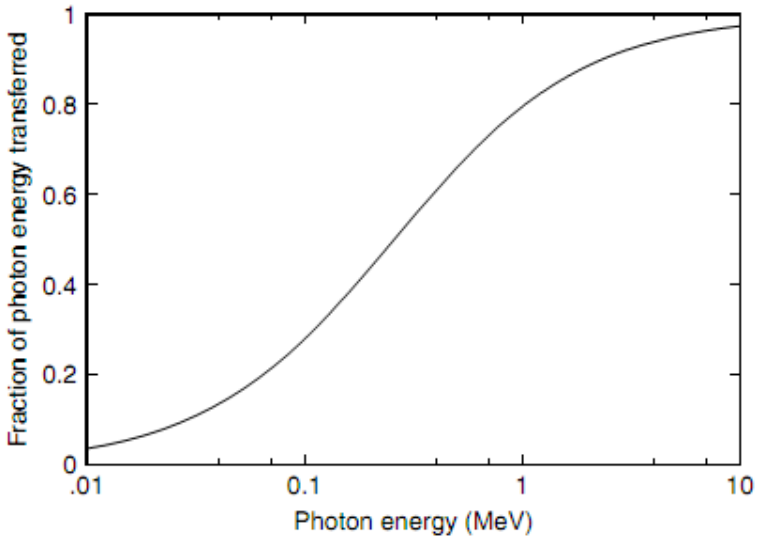

Figure 2.8: Fraction of the incident photon energy that is transferred to the recoil electron for Compton collisions where the photon is backscattered at 180°. (Mayles et al 2007).

The cross-section for incoherent scattering is named after Klein and Nishina who first derived an expression for its value. The differential Klein-Nishina cross-section per electron equals to (Mayles et al 2007):

$$\frac{d\ {}_e\sigma_{KN}(\theta)}{d\vec{\Omega}} = \frac{r_e^2}{2}\left(\frac{h\nu'}{h\nu}\right)^2\left[\frac{h\nu'}{h\nu} + \frac{h\nu}{h\nu'} - \sin^2\theta\right] \qquad (20)$$

where r_e is the classical electron radius given by $r_e = e^2/m_0c^2 = 2.8179 \times 10^{-15}\ m$.

The total Klein-Nishina cross-section per electron is obtained by integrating the differential cross-section, substituting for $h\nu$' using Equation (18), obtaining:

$${}_e\sigma_{KN} = 2\pi r_e\left(\frac{1+\alpha}{\alpha^2}\left[\frac{2(1+\alpha)}{1+2\alpha} - \frac{\ln(1+2\alpha)}{\alpha}\right] + \frac{\ln(1+2\alpha)}{2\alpha} - \frac{1+3\alpha}{(1+2\alpha)^2}\right) \qquad (21)$$

and the total cross-section per atom is ${}_a\sigma_{KN} = Z\ {}_e\sigma_{KN}$.

The energy transferred to the Compton electron will be subsequently deposited close to the point of interaction, within the electron range, whereas, the secondary photon may travel much farther. The differential Klein-Nishina cross-section is shown in Figure 2.9. At low photon energies, the Compton scattering angular distribution is almost symmetrical about $\cos\theta = 0$. With increasing energy the photons are increasingly scattered into the forward direction, and an increasing fraction of the energy is transferred to the Compton electron. At 0.01 MeV only 1.87% of the photon energy is converted to electron kinetic energy. This fraction increases to 50% at 1.7 MeV and reaches 79.4% at 100 MeV (Mayles et al 2007). These fractions are limited by those shown in Figure 2.8.

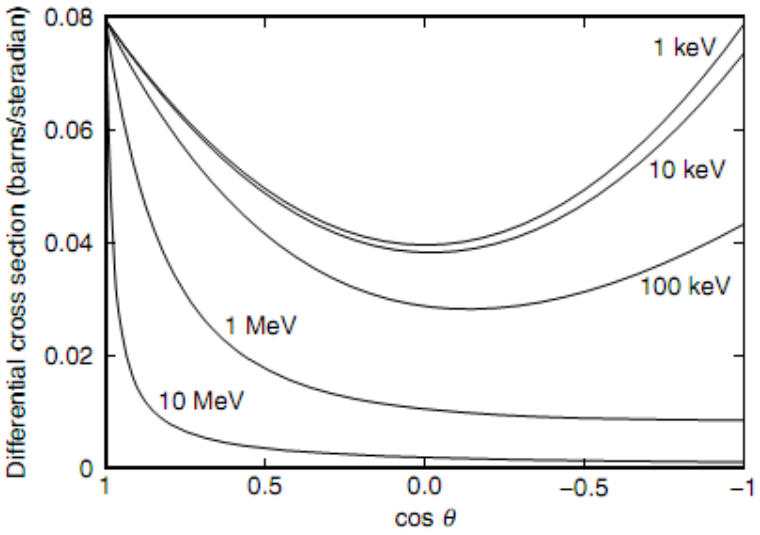

Figure 2.9: Variation of differential cross-section with respect to $\cos\theta$ and the energy of the incident photon. (Mayles et al 2007).

As it was mentioned, the above treatment of the incoherent scattering assumes electrons to be unbounded and initially at rest. In actual case the electrons in atoms have some binding energy depending on the energy level. Also these electrons are in motion. However, the resulting errors from the use of unbounded and stationary assumptions are only minor in radiotherapy applications at high energies. More accurate differential cross sections for incoherent scattering process are obtained by using an impulse approximation, which takes into account the target electron binding effects and the Doppler broadening of the Compton line, caused by the momentum distribution of the target electrons (Kawrakow and Rogers 2003).

2.1.2.3 Coherent (Rayleigh) scattering

In Rayleigh scattering the incident photon is scattered by the combined action for the whole atom. For this reason the process is also called coherent scattering. It is an elastic process, which means that the incident photon does not lose energy in the interaction. The atom moves just enough to conserve momentum and initial photon is redirected in a small angle. The differential cross-section for coherent scattering is (Mayles et al 2007):

$$\frac{d\sigma_{coh}}{d\vec{\Omega}} = \left(\frac{r_e}{2}\right)^2 (1+\cos^2\theta)[F(q)]^2 \qquad (22)$$

where $F(q)$ is the atomic form factor and q is the momentum transfer when a photon with an energy $h\nu$ is scattered by an angle θ, $q = h\nu\sqrt{(1-\cos\theta/2}$. The atomic form factor is a measure of the scattering amplitude of a wave by an isolated atom and depends on the type of scattering, which in turn depends on the nature of the incident radiation (McKie and Mckie 1992). The coherent scattering is more significant at lower energies.

2.1.2.4 Pair and triplet production

In a pair production process a photon is absorbed in the electric field of the nucleus. An electron-positron pair is created and emitted with the sum of their kinetic energies $T^+ + T^-$. The value of the kinetic energy of the particles is derived from the conservation of energy of the interaction:

$$T^- + T^+ = h\nu - 2m_0c^2 \quad (23)$$

Here, m_0c^2 is the energy equivalent of the electron rest mass m_0. The triplet production process is similar but the interaction takes place with the electric field of one of the atomic electrons which receives sufficient energy to be set free.

From the above equation, it is clear that the pair production process has a threshold value of $2m_0c^2$ (1.02 MeV), the minimum energy to create an electron-positron pair. On average, the electron-positron pair about equally shares the kinetic energy available. The probability for either particle to absorb most of the energy is small. In the case of the triplet production an additional electron appears as a result of the interaction and, accordingly, the process has an energy threshold at $4m_0c^2$ (2.04 MeV).

The cross-section for pair production in the nuclear field is zero below this threshold. It then rapidly increases with increasing energy and, well above threshold, varies approximately as the square of the nuclear charge Z, i.e. (Mayles et al 2007):

$$_a\sigma_{pair} \propto Z^2 \quad (24)$$

The cross-section for triplet production, at energies above threshold, approximately varies as Z:

$$_a\sigma_{triplet} \propto Z \quad (25)$$

Triplet production is as important as pair production in hydrogen (Z=1), but it gets increasingly less important, compared to pair production, with increasing atomic number. In high atomic number media, Z dependence of these processes becomes weaker due to screening of the electric fields of the target entities by the surrounding atomic electrons.

2.1.3 Interaction of electrons with matter

There are primarily three interaction mechanisms of importance for electrons in the energy range from a few hundred eV up to 50 MeV: collisions with bound atomic electrons (Møller scattering), elastic scattering and bremsstrahlung or radiative loses. The bremsstrahlung process dominates at high energies whereas the inelastic scattering dominates at low electron energies in high Z media. In tissue-like materials the inelastic scattering dominates at all energies and the influence of the bremsstrahlung process is only a few percent.

The total attenuation coefficients of these processes in water are depicted in Figure 2.10. Because electrons are charged particles, their interaction probabilities are much larger than those of uncharged particles. An electron would typically undergo about hundreds of thousands of interactions before losing all its energy. This supposes a great challenge for computational resources in the case of electron transport simulation (Kawrakow and Bielajew 1998).

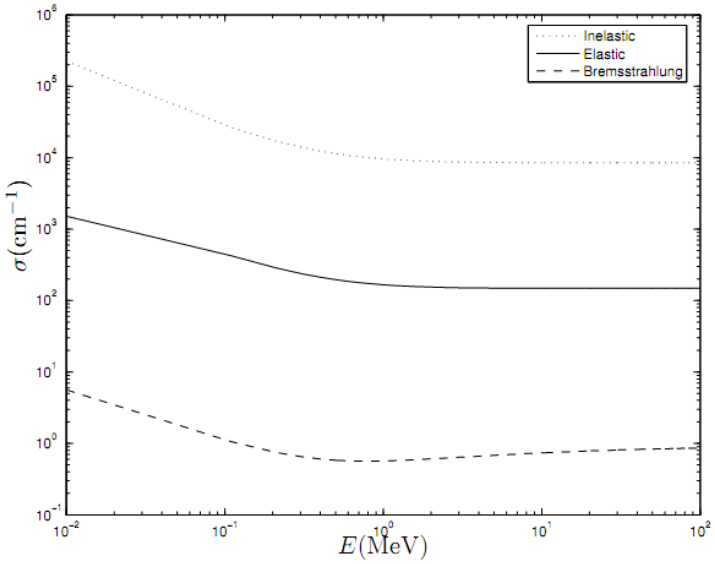

Figure 2.10: The total attenuation coefficients for different electron interactions in water with respect to energy (Boman 2007).

In all cases the consequences of these interactions are twofold: modifications of the incident (direct or secondary) charged particles in terms of energy loss and direction and transfer of energy to matter, resulting in energy absorption and dose deposition.

2.1.3.1 Inelastic electron scattering

Inelastic electron scattering occurs when electrons inelastically interact with atomic electrons ejecting them from the atom with considerable kinetic energy (Attix 1986). The result of this process, also called electron-electron scattering, is that the incident electron scatters producing a secondary electron. When the binding of atomic electrons is ignored, electron-electron scattering can be described by the Møller cross section.

The Møller cross section, which is the cross section for electron-electron scattering differential in the kinetic energy T' of the scattered electron, initially at rest, is (Kawrakow and Rogers 2003)

$$\frac{d\sigma_{inel}^{-}}{dT'} = \frac{2\pi r_0^2 m}{\beta^2}\frac{1}{T'^2}\left[1 + \frac{T'^2}{(T-T')^2} + \frac{\tau^2}{(\tau+1)^2}\left(\frac{T'}{T}\right)^2 - \frac{2\tau+1}{(\tau+1)^2}\frac{T'}{(T-T')}\right] \qquad (26)$$

where β is the incident electron velocity in units of the speed of light, T the incident kinetic energy and $\tau = T/m$. The Møller cross section takes into account relativity, spin effects and the fact that the two scattered particles are indistinguishable (Møller 1932).

Per definition, the electron with the higher energy after the collision is considered to be the primary. Therefore the total cross section for Møller interactions is obtained via integration of Equation (26) from $[T_{max}/2\,, T_{max}]$, where T_{max} is the maximum primary electron energy.

2.1.3.2 Radiative loses (Bremsstrahlung)

The acceleration of the (very light) electrons in the strong electric field of a nucleus leads to the production of bremsstrahlung. In that process a photon is emitted and the initial electron gives significant amount of its kinetic energy to this photon and slows down. The acceleration is proportional to Z/m, where m is the mass of the moving particle and the intensity of radiation produced is then proportional to $(Z/m)^2$. It can be seen that this is a relatively

unimportant energy loss mechanism below about 10 MeV in low-Z materials, and it is completely negligible for heavy charged particles.

The cross-section σ_{brem} for this totally non-classical process is extremely complicated. Usually the original paper of Koch and Motz (1959) is used as a reference to obtain the differential cross-section in emitted photon energy $d\sigma_{brem}/dhv$.

2.1.3.3 Elastic electron scattering

When a charged particle passes close to the atomic nucleus, at a distance much smaller than the atomic radius, the Coulomb interaction will now be between the fast particle and the nuclear charge rather than with one of the bound electrons. In the case of electrons this causes appreciable changes of direction, but almost never (with the exception of the bremsstrahlung process) any change in energy. The scattering is basically elastic, being the energy lost the negligible amount required to satisfy momentum conservation between the very light electron and the nucleus that is at least thousand times heavier (Mayles et al 2007).

This interaction process is essentially a Rutherford scattering. The cross section differential in the cosine of the polar angle θ of the electron incident on atoms of atomic number Z is (Kawrakow and Rogers 2003)

$$\frac{d\sigma_{RS}}{d\,(\cos\theta)} = \frac{2\pi {r_0}^2 Z^2}{\beta^2 \tau(\tau+2)} \frac{1}{(1-\cos\theta+2\eta)^2} \tag{27}$$

where β is the particle velocity in units of the speed of light, $\tau = T/m$, where T and m are the kinetic energy and the mass of the incident particle, and η the screening parameter.

The screened Rutherford scattering formulation is adequately accurate in tissue-like media at energies above 100 keV. More accurate models for electron elastic scattering at low energies can be achieved by taking into account the spin effects with the help of partial wave analysis (Kawrakow and Rogers 2003).

2.2 Particle transport and the Boltzmann transport equation

2.2.1 Introduction

Particles like photons or electrons travel in a medium and interact with medium atoms via different interactions. This particle transport in a medium can be described by the coupled time independent linear Boltzmann Transport Equation (BTE), which is an integro-differential equation and coupled due to the fact that particles change into other types of particles in different interactions.

It was first derived by Ludwig Boltzmann for dilute gases in 1872 from the Liouville equation with appropriate approximations. Although it was generated for ideal gases, it is still used as a basis in nowadays gas theory and adopted from there to the other fields of particle physics. In radiation therapy, the interactions of particles with each other are negligible and the linear BTE can be used to describe the evolution of particles in a medium (Börgers 1999)

2.2.2 Derivation of the Boltzmann transport equation for photons, electrons and positrons

BTE is basically a balance equation. The linear BTE can be derived from the particle conservation within a small volume element of a phase space (Boman 2007). The BTE is derived for the coupled system, including photons, electrons and positrons, assuming particle

conservation. The particles are assumed to interact only with medium atoms, not with each other and the particles are also assumed to travel with straight lines between the interactions.

Due to particle conservation the number of particles within a small volume element $\mathrm{d}V$ is:

$$dN_j = -dN_{j,in-out} - dN_{j,att} + dN_{j,sec} + dN_{j,source} \tag{28}$$

where the following definitions apply:

$dN_{j,in-out}$	the net number of particles j flowing out of the volume $\mathrm{d}V$.
$dN_{j,att}$	the number of particles j that are attenuated in $\mathrm{d}V$.
$dN_{j,sec}$	the number of secondary particles j that are born in the scattering interactions with medium atoms in $\mathrm{d}V$.
$dN_{j,source}$	the number of particles j produced by sources inside the volume $\mathrm{d}V$.
j	specifies the particles, $j = 1,2,3$ referring to photons, electrons and positrons, respectively.

Consider the Figure 2.11. Using Equation (7) the net number of particles j flowing out of the volume dV at the surface element dA is $\vec{\Omega}\Phi_{j,E,\Omega}d\vec{\Omega}dEd\vec{A}$, where $\Phi_{j,E,\Omega} = d\Phi_j/dEd\vec{\Omega}$ is the fluence of particles j differential in energy and solid angle and $d\vec{A} = \hat{n}dA$ is the normal vector to the surface element dA.

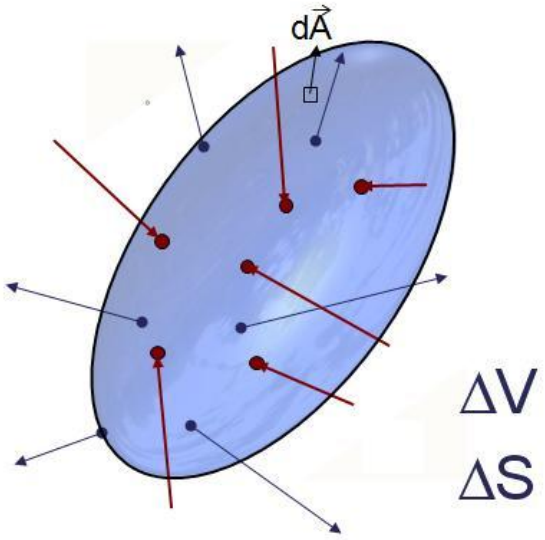

Figure 2.11: Flux of particles through a small volume $\mathrm{d}V$ with surface dS.

Therefore, the number of particles j flowing out of the volume $\mathrm{d}V$ through the entire surface dS is:

$$dN_{j,in-out} = \int_{dS} \vec{\Omega}\Phi_{j,E,\Omega}d\vec{\Omega}dEd\vec{A} \tag{29}$$

Applying the Gauss Theorem, Equation (29) can be written as a volume integral:

$$dN_{j,in-out} = \int_{dV} \vec{\Omega}\nabla\Phi_{j,E,\Omega} d\vec{\Omega} dE dV \tag{30}$$

The number of particles j that are attenuated in dV is obtained introducing $\sigma_{j,att}$ as the probability per unit path length L_j for particle j of energy E to attenuate. Therefore we obtain that:

$$dN_{j,att} = \int_{dV} \sigma_{j,att}\, dL_j = \int_{dV} \sigma_{j,att} \Phi_{j,E,\Omega} d\vec{\Omega} dE dV \tag{31}$$

where Equation (8) was used to write dL_j in terms of the particle j fluence $\Phi_{j,E,\Omega}$.

The number of particles j that are born in the scattering interactions with medium atoms in dV is calculated introducing $\sigma_{j'-j}$ as the probability per unit path length that a particle j' with energy E' and direction $\overrightarrow{\Omega'}$ will produce a secondary particle j with energy E and direction $\vec{\Omega}$. Hence it follows:

$$\begin{aligned} dN_{j,sec} = \int_{dV} \int_{\vec{\Omega}} \int_{E} \sum_{j'=1}^{3} \sigma_{j'-j} dL_{j'} dE' d\overrightarrow{\Omega'} \\ = dE d\vec{\Omega} \int_{dV} \int_{\vec{\Omega}} \int_{E} \sum_{j'=1}^{3} \sigma_{j'-j} \Phi_{j',E',\Omega'} dE' d\overrightarrow{\Omega'} \mathrm{dV} \end{aligned} \tag{32}$$

A source inside the volume dV is described through the source term $Q_j(\vec{r}, \vec{\Omega}, \mathrm{E})$ for particle j. Therefore the number of particles j produced by the source inside the volume dV corresponds to:

$$dN_{j,source} = \int_{dV} Q_j d\vec{\Omega} dE dV \tag{33}$$

Replacing Equations (30), (31), (32) and (33) in Equation (28) is obtained:

$$\begin{aligned} dN_j = d\vec{\Omega} dE \int_{dV} \Bigg[-\vec{\Omega}\nabla\Phi_{j,E,\Omega} - \sigma_{j,att}\Phi_{j,E,\Omega} \\ + \int_{\vec{\Omega}} \int_{E} \sum_{j'=1}^{3} \sigma_{j'-j} \Phi_{j',E',\Omega'} dE' d\overrightarrow{\Omega'} + Q_j \Bigg] dV \end{aligned} \tag{34}$$

After having exposed the volume of interest an equilibrium state is reached, this means that the number of particles entering the volume element is the same as the number of particles leaving it. Therefore the integrand in Equation (34) must be zero, i.e.:

$$-\vec{\Omega}\nabla\Phi_{j,E,\Omega} - \sigma_{j,att}\Phi_j \int_{\vec{\Omega}} \int_{E} \sum_{j'=1}^{3} \sigma_{j'-j} \Phi_{j',E',\Omega'} dE' d\overrightarrow{\Omega'} + Q_j = 0 \tag{35}$$

Finally a set of equations is obtained for photons, electrons and positrons ($j = 1,2,3$ respectively):

$$-\vec{\Omega}\nabla\Phi_{1,E,\Omega} - \sigma_{1,att}\Phi_{1,E,\Omega} \int_{\vec{\Omega}} \int_{E} \sum_{j'=1}^{3} \sigma_{j'-1} \Phi_{j',E',\Omega'} dE' d\overrightarrow{\Omega'} + Q_1 = 0 \tag{36}$$

$$-\vec{\Omega}\nabla\Phi_{2,E,\Omega} - \sigma_{2,att}\Phi_{2,E,\Omega}\int_{\vec{\Omega}}\int_{E}\sum_{j'=1}^{3}\sigma_{j'-2}\Phi_{j',E',\Omega'}dE'd\vec{\Omega'} + Q_2 = 0 \qquad (37)$$

$$-\vec{\Omega}\nabla\Phi_{3,E,\Omega} - \sigma_{3,att}\Phi_{3,E,\Omega}\int_{\vec{\Omega}}\int_{E}\sum_{j'=1}^{3}\sigma_{j'-3}\Phi_{j',E',\Omega'}dE'd\vec{\Omega'} + Q_3 = 0 \qquad (38)$$

These set of equations are well known as the Boltzmann transport equation, which are based on the conservation of particles in space. It represents a coupled integro-differential system of stationary linear equations for external radiation therapy. By solving this equations system the fluence of electrons and positrons can be obtained and hence the absorbed dose imparted by these particles.

2.3 The Monte Carlo simulation of electromagnetic radiation

As applied to radiotherapy and dosimetry, the Monte Carlo method provides a solution method to the Boltzmann transport equation that can be applied for any energy range of interest (Duderstadt and Martin 1979). Monte Carlo simulations of particle transport processes are a faithful simulation of physical reality: particles are born according to distributions describing the source, they travel certain distances, determined by a probability distribution depending on the total interaction cross section, to the site of a collision and scatter into another energy and/or direction according to the corresponding differential cross section, possibly producing new particles that have to be transported as well (Kawrakow and Rogers 2003). The simulation tracks the particle trajectories in the 6D phase space room (i.e. position, direction and energy of the particle) and scores the track-lengths of the particles in a volume of interest inside the geometry that is being simulated. Quantities of interest (e.g. absorbed dose) are computed as an average over many individual particle simulations or histories. As such, the Monte Carlo estimate of quantities of interest is subject to a statistical uncertainty which depends on N, the number of particle histories simulated, and usually decreases as $N^{1/2}$. If the true average $\bar{x}$ exist and the distribution in x has a true finite variance, σ_x^2, the Central Theory Limit Theorem (Lindeberg 1922; Feller 1967) for energies 0.001 MeV up to 50 MeV guarantees that the Monte Carlo estimator $\langle x\rangle$ for $\bar{x}$ can be made arbitrarily close to $\bar{x}$ by increasing the number N of particle histories simulated. Moreover, the Central Limit Theorem predicts that the distribution of $\langle x\rangle$ is Gaussian, characterised by a variance $\sigma_{\langle x\rangle}^2$ that may be simply estimated in the simulation. The Central Limit Theorem also predicts that in the limit $N \to \infty$,$\sigma_{\langle x\rangle}^2 \to 0$. This limiting result is also proven by the Strong Law of Large Numbers (Feller 1967).

It is these facts that have been partly responsible for the rapid increase in the use of the Monte Carlo method in radiotherapy and dosimetry applications. If one knows the governing physical laws to sufficient accuracy and has access to sufficient computing resources, then the answer to any well-posed physical question may be computed. Fortunately, the physical laws required for most applications in radiotherapy and dosimetry are well known (Bjorken and Drell 1965, Sakurai 1967). Additionally, the computer resources required for most of our applications are modest, and they may be executed to sufficient accuracy on affordable desktop computers and workstations. This confluence of theory and computational ability puts the Monte Carlo method more and more into the standard toolbox of the medical physicist, especially if they are involved in research. As it was stated already in the introduction, this work is considered as a contribution to this process.

2.3.1 Monte Carlo Method

One of the best examples to introduce the Monte Carlo method is also the oldest associated with its underlying principle (random sampling). This is known as “Buffon’s needle problem” after the French naturalist who first described it in the eighteenth century (Raeside 1976). Buffon proposed a Monte Carlo-like method to evaluate the probability of tossing a needle onto a ruled sheet. He calculated that a needle of length L randomly tossed on a plane ruled with parallel lines of distance d apart, where $d > L$, would have a probability of intersect one of the lines equal to $p = 2L/\pi d$. If the needle is thrown N times and if n intersections are observed, then $\pi \cong (2L/d)(N/n)$, where N is a large number. This idea can serve as the bases for a modern Monte Carlo calculation of π using a digital computer. In a computer simulation of the random throwing of the needle the computer must generate two independent uniformly distributed random numbers, one to be used in specifying the location of the needle’s centre and the other to specify its inclination angle. In addition, the computer program must contain an intersection criterion, that is, a criterion which will serve as a test to detect the cases where an intersection occurs. The computer also must keep tallies of the number of throws and the number of intersections so that the product $(2d/L)(N/n)$ can be formed as an output of the program. The modern Monte Carlo age was ushered by von Neumann and Ulam during the initial development of thermonuclear weapons. Ulam and von Neumann coined the phrase Monte Carlo, and they were pioneers in the development of the Monte Carlo technique and its realizations on digital computers.

The following subsections describe the key elements of the Monte Carlo method as used to solve the BTE.

2.3.1.1 Pseudo Random number generators

Monte Carlo calculations require large supplies of high quality random numbers. We shall understand a random number to be a particular value of a continuous random variable distributed on the unit interval. The quality of a supposedly random sequence of number can be established only after a very careful analysis aimed at discovering patterns in random number sequences. We portray a random number sequence as a vague notion embodying the idea that each number of the sequence is unpredictable and that the sequence’s stream of digits passes certain statistical tests (Raeside 1976).

Through the years a number of methods of obtaining random numbers have come into use, where today the calculation using a specified mathematical algorithm is the principal method of obtaining random numbers. Due to this deterministic production of “random” numbers using mathematical algorithms the term “pseudo random” is coined (Raeside 1976). One important shortcoming, however, is associated with the use of all pseudo random number generators. It may happen that after a certain number of distinct elements have been produced the sequence begins to repeat itself, the number of distinct digits produced being called its period. If the period of a sequence is very large this periodic behavior is of no practical importance for Monte Carlo calculations. The period of the RANLUX pseudo random number generator used in the EGSnrc platform is about 10^{171} (Lüscher 1994; James 1994).

The pseudo random number generator is the soul of a Monte Carlo calculation. It is what generates the pseudo random nature of Monte Carlo simulations thereby imitating the true stochastic nature of particle interactions.

2.3.1.2 Elementary sampling theory

The aim of any Monte Carlo procedure is to draw independent random samples from some probability law via intermediate steps involving the use of independent random numbers. There are a number of sampling techniques well known to all constructors of Monte Carlo codes, of which the direct (or inversion) method and the rejection method will be described below.

2.3.1.2.1 Direct (inversion) method

The basic idea behind the direct method of random sampling is simple. Assume that a probability density function $PDF(x)$ that is normalized over some range between a and b is used, that is:

$$\int_a^b dx' PDF(x') = 1 \tag{39}$$

Its cumulative probability distribution function $CDF(x)$ is now constructed:

$$CDF(x) = \int_a^x dx' PDF(x') \tag{40}$$

Consistent with the probability theory, $0 < CDF(x) < 1$. It can be shown that a variable x is distributed randomly according to $PDF(x)$ if x values are selected by inverting $CDF(x)$ according to $x = CDF^{-1}(R)$, where R is a uniformly distributed random number between 0 and 1.

As an application for the direct method of Monte Carlo sampling, consider the problem of generating the random distances which photon travel between interactions (Raeside 1976). The probability that no interaction occurs for a photon travelling a distance x is $\exp(-\mu x)$, where μ is the total attenuation coefficient of the medium and $\mu \exp(-\mu x)\, dx$ is the probability an interaction occurs in the distance x to $x + dx$. The appropriate cumulative probability is

$$CDF(x) = \int_0^x \mu \exp(-\mu x)\, dx = 1 - \exp(-\mu x) \tag{41}$$

A random sample x_R can be obtained by generation of a random number R $(0 < R < 1)$ and solving the equation $R = 1 - \exp(-\mu x_R)$ for R:

$$x_R = -\left(\frac{1}{\mu}\right) \ln(1 - R) \tag{42}$$

Since $0 < (1 - R) < 1$, and since $1 - R$ has the same probability distribution as R, one can use the equivalent form of the above equation $x_R = -(1/\mu)\ln(R^*)$.

If more than one interaction process comprises μ, then another random number R_2 must be selected to choose which interaction occurs. The total attenuation coefficient, independent of which process occurs, is the sum of the attenuation coefficients for the individual processes. For example if only incoherent scattering and photo-electric absorption are considered the total attenuation coefficient is equal to

$$\mu = \mu_{photoelectric} + \mu_{\text{incoherent}} \tag{43}$$

Therefore, if $R_2 > \mu_{\text{incoherent}}/\mu$ then the process is photo-electric, otherwise it is incoherent scattering. This is the form used to calculate particles mean free paths in the EGS platform.

2.3.1.2.2 Rejection method

While the direct method is always possible, at least in principle, it is often impractical to calculate $CDF^{-1}(R)$ because it may be exceedingly complicated mathematically. Another approach is to use the rejection sampling technique. The name has its roots in the fact that not all of the random samples generated are retained; some are rejected on the basis of a specified criterion (Raeside 1976).

The probability density function is scaled by its maximum value obtaining $PDF' = PDF/PDF_{max}$, which has a maximum value of 1. A representation of $PDF'(x)$ is shown in Figure 2.12. Clearly this method works only if the PDF is not infinite anywhere and if it is not prohibitively difficult to determine the location of the maximum value. A random number r_1 ($0 < r_1 < 1$) is used to obtain an x which is uniform in the PDF's range using the expression $x = a + (b - a)r_1$. It can be seen that this method is restricted to finite values of a and b. However, if either a or b are infinite a suitable transformation may be found to allow one to work with a finite range. Finally, a second random number r_2 is chosen. If $r_2 < PDF(x)/PDF_{max}$ (region under $PDF'(x)$ in Figure 2.12) then x is accepted, otherwise it is rejected (shaded region above $PDF'(x)$) and a new x must be obtained. This method will result in x being selected according to the probability density function (Bielajew 1993).

Although in this method random numbers are "wasted" (specially for "spiked" $PDF'(x)$; consider the shaded rejection area in Figure 2.12, meaning that a much larger number of random numbers are needed to obtain x) it can save computing time if the CDF^{-1} is very complicated.

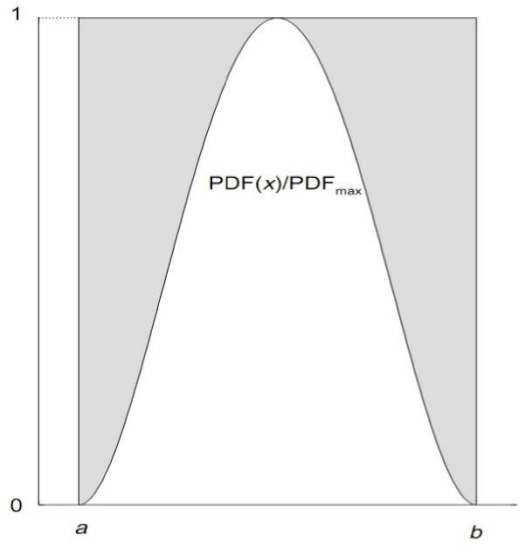

Figure 2.12: The scaled probability distribution $PDF'(x) = PDF(x)/PDF_{max}$ used in the rejection technique (Bielajew 1993).

2.3.1.2.3 Mixed Method

Finally, one can consider a "mixed method", a combination of the previous two methods. Imagine that the PDF is to difficult to integrate and invert, ruling out the direct approach without a big deal of numerical analysis, and that it is "spiky", rendering the rejection method inefficient. However, imagine that the PDF can be factored as $PDF(x) = f(x) \cdot g(x)$, where $f(x)$ is an invertible function that contains most of the "spikiness", and $g(x)$ is relatively flat but contains most of the mathematical complexity. Therefore is method is as follows (Bielajew 1993):

1. $f(x)$ is normalized producing $\tilde{f}(x)$ in such a way that $\int_a^b \tilde{f}\,(x) = 1$.
2. $g(x)$ is normalized producing $\tilde{g}(x)$ with $\tilde{g}(x) \leq 1\ \forall\ x \in [a, b]$.
3. Using the direct method a x is obtained using $\tilde{f}(x)$ as the PDF.
4. Using this x, the rejection technique is applied using $\tilde{g}(x)$. A random number r $(0 < r < 1)$ is chosen. If $\leq \tilde{g}(x)$, x is accepted, otherwise it is rejected and one goes back to step 3.

As an example of the mixed method of random sampling look at the angular sampling of newly created bremsstrahlung photons in the EGS system (Bielajew et al 1989). For the angular sampling routine the cross section formulated by Koch and Motz (1959), differential in photon energy and angle, is employed:

$$d\sigma_{k,\Theta} = \frac{4Z^2 r_0^2}{137}\frac{dk}{k} y dy \left\{ \frac{16y^2 E}{(y^2+1)^4 E_0} - \frac{(E_0+E)^2}{(y^2+1)^2 {E_0}^2} + \left[\frac{{E_0}^2 + E^2}{(y^2+1)^2 {E_0}^2} - \frac{4y^2 E}{(y^2+1)^3 E_0} \right] \ln M(y) \right\} \quad (44)$$

where,

$$y = E_0 \Theta;\ \frac{1}{M(y)} = \left(\frac{k}{2E_0 E}\right)^2 + \left(\frac{Z^{1/3}}{111(y^2+1)}\right)^2$$

and the following definitions for the variables apply:

k energy of the photon in units of $m_e c^2$.

Θ angle between the outgoing photon and the incoming electron direction.

r_0 $\equiv e^2/m_e c^2$ classical electron radius.

E_0, E initial and final electron energy in units of $m_e c^2$.

To sample the photon angular distribution a mixed sampling procedure is employed. Since it is the angular distribution that is required, the overall normalization of Equation (44) is unimportant including any overall energy-dependent factors. The following expression for the probability density function $p(y)$ is proportional to it (Bielajew et al 1989):

$$p(y)dy = f(y^2) N_r g(y^2) dy^2 \quad (45)$$

where, using $x = y^2$,

$$f(x) = \frac{1 + 1/(\pi E_0)^2}{(x+1)^2} \tag{46}$$

$$g(x) = 2r - 3(1+\varepsilon^2) - [4 + \ln m(x)][(1+\varepsilon^2) - 4x\varepsilon/(x+1)^2] \tag{47}$$

where,

$$\varepsilon = \frac{E}{E_0}; m(x) = \left(\frac{1-\varepsilon}{2E_0\varepsilon}\right)^2 + \left(\frac{Z^{1/3}}{111(x+1)}\right)^2;\ N_r = {g_{max}}^{-1}$$

The function $f(x)$ will be used for direct sampling. It can be easily verified that this function is normalized correctly (i.e. $\int_0^{(\pi E_0)^2} f(x)dx = 1$) and the candidate scattering angle is easily found by inversion to be:

$$\Theta = \frac{1}{E_0}\sqrt{\frac{r_1}{1 - r_1 + 1/(\pi E_0)^2}} \tag{48}$$

where r_1 $(0 < r_1 < 1)$ is a random number. The function g is sampled using the rejection technique. To justify this prescription, consider the plots of f and g depicted in Figure 2.13 for four incident e kinetic energy, 100 keV, 1MeV, 10 MeV and 1000 MeV for hydrogen (Z=1). Each figure contains six curves for g corresponding to $k = \left(0, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, 1\right) \times k_{max}$, where $k_{max} = E_0 - 1$. From Figure 2.14 one sees the motivation for utilizing the mixed sampling method. The function f (dashed line) is strongly peaked and hence is an unsuitable candidate for the rejection technique. However, it can be inverted, requiring only the calculation of a square root, and therefore, the direct technique may be applied efficiently. The function g is much flatter and is a more reasonable candidate for the rejection technique. Therefore, the following algorithm is employed to sample the angular distribution (Bielajew et al 1989):

1. Pick a candidate Θ using Equation (48).
2. Evaluate $g_{test} = N_r g((E_0\Theta)^2)$.
3. Pick a random number r_2, $0 < r_2 < 1$.
4. If $r_2 \leq g_{test}$ accept Θ, otherwise one goes back to step 1.

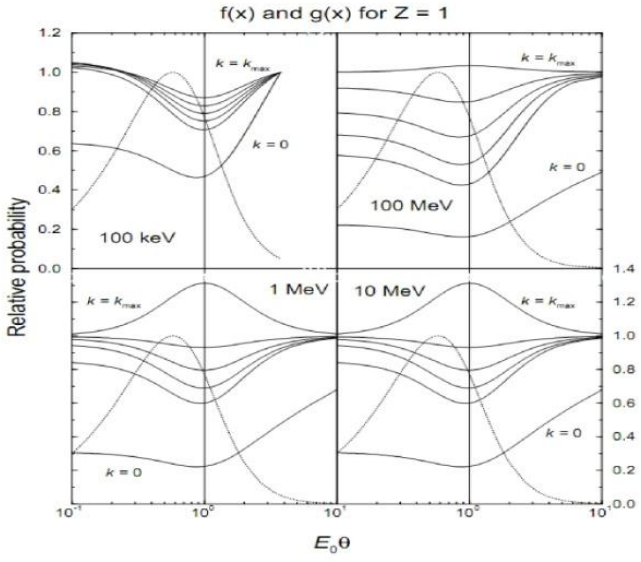

Figure 2.13: The functions f (dashed line) and g (solid lines) for 100 keV, 1 MeV, 10 MeV and 100 MeV for H ($Z = 1$). Six curves for g are depicted for each incident energy corresponding to $k = \left(0, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, 1\right) \times k_{max}$ (Bielajew et al 1989).

2.3.1.3 Condensed History Technique

An important difficulty occurs in the case of the Monte Carlo simulation of electron transport. In the process of slowing down, a typical fast electron undergoes of the order of 10^5-10^6 collisions with surrounding matter (Kawrakow and Bielajew 1998). Because of this large number of collisions, an event-by-event simulation of electron transport is often not possible due to limitations in computing power. To circumvent this difficulty the “condensed history” (CH) technique for the simulation of charged particle transport was developed (Berger 1963). In this method large numbers of subsequent transport and collision processes are condensed to a single “step”. The cumulative effect of the individual interactions is taken into account by sampling the change of the particle’s energy, direction of motion and positions at the end of the step from appropriate multiple scattering distributions. Berger also defined two different implementations of the CH technique, which he called Class I and Class II schemes.

EGSnrc uses a Class II CH scheme for the simulation of electron transport (Kawrakow and Bielajew 1998). Bremsstrahlung processes that result in the creation of photons above an energy threshold E_γ and inelastic collisions that set in motion atomic electrons with kinetic energies above E_δ are simulated explicitly and the secondary particles transported. Such interactions are also referred to as “catastrophic” collisions. Sub-threshold processes are accounted for in a continuous slowing down approximation (CSDA).

A typical Monte Carlo electron track during a simulation is shown in Figure 2.14. Along the way energy is continuously lost and the track is broken up into small straight line segments called “multiple scattering sub steps”. At the end of each of these steps the multiple scattering angle is selected according to some appropriate multiple scattering distribution. Catastrophic events, here a single knock-on electron, set other particles in motion that are followed separately in the same fashion. The original particle is also transported until it falls below the transport threshold (Bielajew and Rogers 1993).

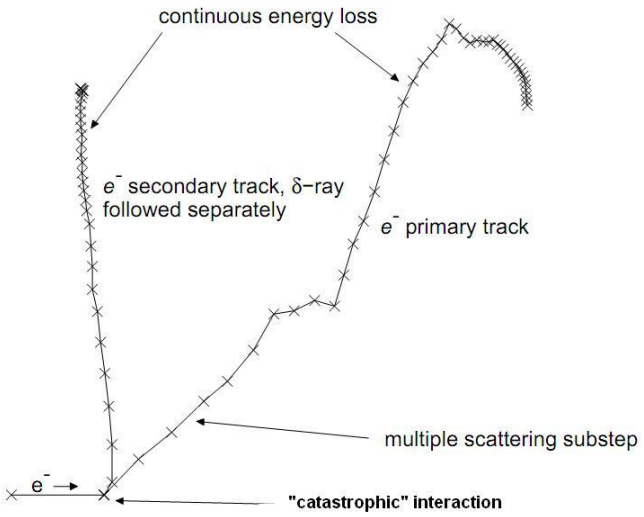

Figure 2.14: A typical electron track during a Monte Carlo simulation (Bielajew and Rogers 1993).

The CH technique, motivated by the fact that single collisions with the atoms cause, in most cases, only minor changes in the particle's energy and direction of flight, made the MC simulation of charged particle transport possible.

2.3.1.4 Variance reduction

When comparing two different Monte Carlo procedures used to estimate the same quantity, it is convenient to assign an efficiency to each. Intuitively, this efficiency should increase as the amount of computer time required for each trial decreases and also should increase as the variance associated with the estimate decreases. Thus, if ε denotes the efficiency of a Monte Carlo procedure then (Raeside 1976)

$$\varepsilon \propto \frac{1}{t\sigma^2} \tag{49}$$

where t is the amount of time required to complete one Monte Carlo trial and σ^2 is the sampling variance associated with the estimate of the quantity of interest.

Variance reduction is the most important modern aspect of the Monte Carlo theory. To illustrate this importance consider the calculation of π by computer simulation of the Buffon needle problem (Raeside 1976). If that simulation is depicted by plotting a graph with the estimate of π obtained after N throws of the needle as the ordinate and N as the abscissa, it would be clearly evident that the rate of convergence to the true value of π deteriorates as N becomes larger and larger. This behavior is such that for each additional decimal digit obtained in the estimate of π, 100 times as much computing must be performed as for the preceding decimal digit. All Monte Carlo calculations suffer from such problem. By applying variance reduction, however, this fact can be compensated and the necessity of accumulating massive amounts of data to achieve a certain precision in the quantity being estimated can be avoided.

One of the variance reduction methods used in the EGSnrc platform is the technique called "range rejection". In this case particles that are not able to reach a certain region of interest in the simulation geometry are discarded. For example, a particle detector may contain a sensitive volume where one wishes to calculate the energy deposition, or some other quantity. Surrounding this sensitive volume may be shields, converters, walls, etc. where one

wants accurate particle transport to be accomplished but where one does not need to score quantities directly. Electrons that cannot reach the sensitive volume may be discarded on the spot, providing that the neglect of the bremsstrahlung photons causes no great inaccuracy. Therefore there is no time wasted in particles that will not reach the region of interest, the variance is maintained and the efficiency of the MC calculation is increased. Or in another way, expending the same time more particles could be simulated reducing the variance of the estimation of the quantity of interest.

2.3.1.5 Statistical analysis

2.3.1.5.1 Scoring variable

A scoring variable x is a quantity calculated during the course of a Monte Carlo simulation (e.g. the absorbed dose in a medium). The output of a Monte Carlo simulation is usually useless if the uncertainty related with the calculation is not given. The conventional approach to calculate the uncertainty in the EGSnrc platform depend on splitting calculations into statistical batches and then, once the simulation is finished, taking the estimate of the uncertainty in the average of the scored quantity.

If the simulation calls for N "incident" particle histories splited in n statistical batches of N/n particles each and for each of these batches the value x_i, $1 \leq i \leq n$, is scored then the mean value of x is:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{n} x_i \tag{50}$$

The unbiased estimator of the variance of x is equal to:

$$s_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}\sum_{1}^{n}\left(x_i^2 - \bar{x}^2\right) \tag{51}$$

The estimated variance of $\bar{x}$ is the variance of the mean:

$$s_{\bar{x}}^2 = \frac{s_x^2}{n} \tag{52}$$

Finally, the result for the scoring variable x is reported as $x = \bar{x} \pm s_{\bar{x}}$.

2.3.1.5.2 Derived quantities

In order to obtain the uncertainty of a derived quantity (i.e. a quantity that is calculated from the scoring variables obtained during a Monte Carlo simulation) the specifications of the Joint Committee for Guides in Metrology were followed (JCGM/WG1 2008).

If the quantity y is obtained from the scoring variables x_i, $1 \leq i \leq N$ (i.e. $y = f(x_i)$), then the uncertainty of y is obtained by appropriately combining the uncertainties of the input variables x_i. This combined uncertainty is denoted $u_c(y)$ and it is equal to:

$$u_c^2(y) = \sum_{i=1}^{N}\left(\frac{\partial f}{\partial x_i}\right)^2 u^2(x_i) \tag{53}$$

where $u(x_i)$ is the uncertainty of the scoring variable x_i. The combined uncertainty $u_c(y)$ is an estimated standard deviation and characterizes the dispersion of the values that could reasonably be attributed to the quantity y. If Equation (53) is normalized with respect to y^2 the following expression is obtained:

$$\left(\frac{u_c(y)}{y}\right)^2 = \sum_{i=1}^{N} \left(\frac{\partial f}{\partial x_i}/y\right)^2 u^2(x_i) \quad (54)$$

In the particular case of $y = f(x_1, x_2) = x_1 \times x_2$ or $y = f(x_1, x_2) = x_1 \div x_2$ it is easy to proof that the combined uncertainty of y is:

$$u_c^2(y) = \left(\frac{u(x_1)}{x_1}\right)^2 + \left(\frac{u(x_2)}{x_2}\right)^2 \quad (55)$$

As an example of the calculation of a combined uncertainty, take the calculation of the off-axis ratio (OAR) for a certain beam using a Monte Carlo simulation. The off-axis ratio is usually defined as the ratio of dose at an off-axis point to the dose on the central beam axis at the same depth in a phantom (Podgorsak 2005), therefore:

$$OAR(x) = \frac{D(x)}{D_{ref}} \quad (56)$$

where $D(x)$ is the dose at position x of the central axis and D_{ref} is the dose at the central axis (i.e. $D_{ref} = D(0)$). In a Monte Carlo simulation the dose at each off-axis position x is obtained, so $D(x)$ and D_{ref} are the scoring variables. Using Equation (55) the combined uncertainty for the off-axis ratio at x is equal to:

$$u_c^2(OAR(x)) = \left(\frac{u(D(x))}{D(x)}\right)^2 + \left(\frac{u(D_{ref})}{D_{ref}}\right)^2 \quad (57)$$

where $u(D(x))$ and $u(D_{ref})$ are the estimated uncertainties of $D(x)$ and D_{ref} during the course of the Monte Carlo simulation.

2.3.2 Example to determine the history of photons

Figure 2.15 is a schematic illustration of an actual photon history that includes the tracks of the secondary particles. The history begins at position 1 in a vacuum; photons are indicated as sinusoidal tracks and the secondary electrons by straight lines. The simulation consists of the following steps, labelled explicitly in the figure (Mayles et al 2007):

1. Choose photon energy, direction and starting position based on sampling from distribution of incident photons, and transport the photon to first boundary.
2. Obtain the distance to first interaction by sampling from the exponential probability law and transport photon to this interaction point.
3. Determine the type of interaction (Compton scatter, photo-electric, pair production, Rayleigh scatter), based on the interaction cross sections of each process.
4. Calculate direction, energy, etc. of new particles (such as Compton electrons by sampling from the Klein-Nishina differential cross-section; characteristics photons; Auger electrons). Put them on the stack (i.e. a list of secondary particles to be followed later).
5. Transport scattered photon until it either leaves the geometry, or it reaches some predetermined energy cut-off (e.g. PCUT in the EGS code system).
6. Transport secondary electron by using the CH method. Keep track of any δ electrons and bremsstrahlung photons produced by putting them on the stack.

7. Score deposited energy, fluence spectra, etc. in region of interest. Estimate the uncertainty of the scored quantities.
8. Repeat steps 1-7 for many more particles until scored quantities reach a pre-defined low statistical uncertainty.

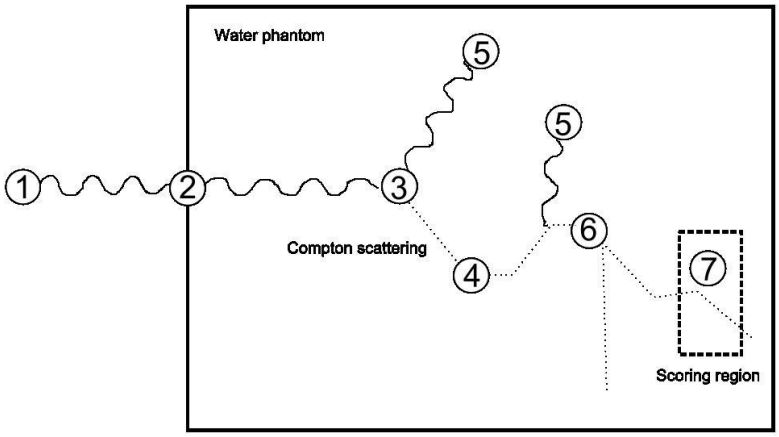

Figure 2.15: Schematic illustration of a Monte Carlo photon electron transport (Mayles et al 2007).

2.4 DKFZ Monte Carlo Platform

2.4.1 General aspects

The DKFZ platform was established based on the existing EGSnrc platform with the following characteristics:

- Use of MORTRAN for code sourcing is maintained. Thus the bridge to the original EGSnrc platform is preserved.
- The MAKEFILE method is applied to build a FORTRAN code only. In the original platform this method is used to generate the executable code too. The content of a MAKEFILE is thus substantially reduced compared to the original one and also due to the fact that it was made fitting to our environment only. It essentially contains a direct and complete list of all required MORTRAN and MACRO files including their locations and the instruction to build a FORTRAN code.
- A simple folder structure was created. The main folder of the new platform is called Monte_Carlo_Directory (see Figure 2.16). The files needed to build the EGSnrc codes are reordered in such a way that all files required for a particular code are put closely together.
- Names and location of all required input data files for a particular MC code are put together in a particular file called data_file_handling. Any input file (except this file itself) is read in from this data handling file. This enables a good control of the entire data flow into and out of a particular MC code.
- Code modifications, amendments or new developments are all performed in the FORTRAN source file via the Microsoft Visual Studio using the IVFC. Following this, they are back projected into MORTRAN source files.

Thus the building of a particular code such as the BEAMnrc code to simulate a radiotherapy source follows essentially the same rules as described in the original BEAMnrc User's Manual (Kawrakow and Rogers 2003).

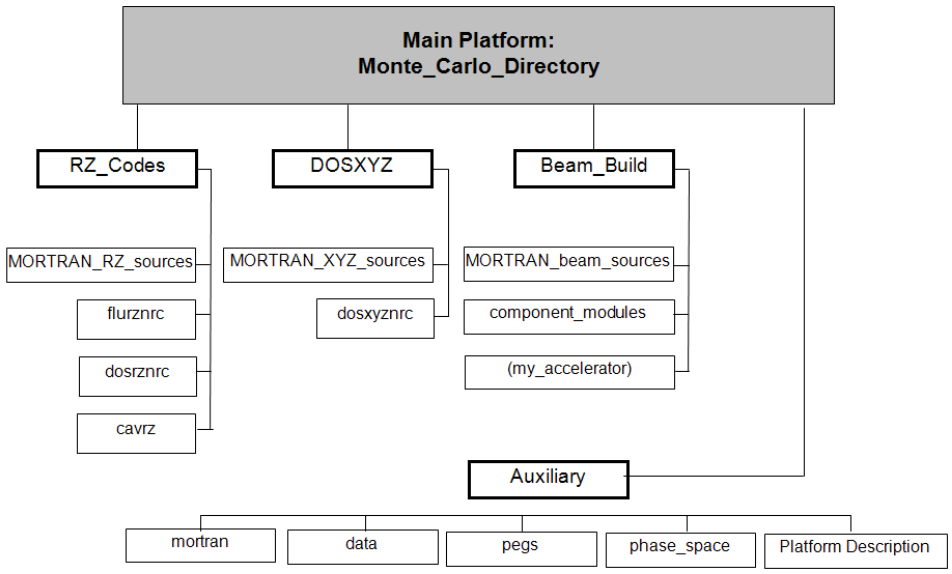

Figure 2.16: Current structure and location of folders and files in the DKFZ platform.

2.4.2 Specific aspects related to the BEAMnrc code

All the files needed in order to develop a simulation of a radiotherapy source are located in specific subfolders.

The beam_build.f program, a modified FORTRAN subroutine that creates the files needed to build and run the program of a desired radiotherapy source, is placed in beam_build folder, together with the my_accelerator.module file, where the component modules that define the radiotherapy source are listed (Figure 2.16).

The main codes taken from the original platform are located in MORTRAN_beam_sources folder. The material dependent cross section data sets needed for the simulation of photon and electron interactions are stored in data and pegs4 folders, both inside the auxiliary folder. The MORTRAN codes of the component modules are located in component_modules folder.

2.4.2.1 Procedures to build a code for accelerator simulation

The building of the code for a radiotherapy source consists of three steps. First, the code generating program beam_build.f is executed. This creates a subfolder in the beam_build folder called my_accelerator. The MAKEFILE needed for the simulation of the radiotherapy accelerator is created inside this folder. Therefore, the MAKEFILE must be processed via cmd.exe (on Windows) with the command make. After that, the file beam_my_accelerator.f is created, which correspond to the radiotherapy accelerator program. After compilation, it can be executed directly as well as in the debugging mode using the IVFC.

The inputs needed by the program to run the simulations are given by a beam_my_accelerator.egsinp file, like in the original BEAMnrc platform. This file is created by the user and must be placed in my_accelerator folder. Once the program is executed with the IVFC, the phase space file is created in my_accelerator folder.

2.4.2.2 The phase space file

After a simulation of a radiotherapy source the phase space file scored during the simulation is obtained. This file contains phase space information (particle position, direction, charge, energy, etc.) for all particles crossing the scoring planes defined in the input file.

This phase space file can be then used as a source for a further Monte Carlo simulation using BEAMnrc or other of the programs in the EGSnrc platform, like DOSXYZnrc. For example, consider a model of a ^{60}Co unit which is broken into two components. In the first part the source capsule is modeled and a phase space file is created with all the particles leaving the surface of the capsule. In the second stage, the phase space file from the first part is used as an input to a model of the collimator system and a second phase space file is scored. Finally, this file, which characterizes the radiation output of the radiotherapy unit can be used as an input for dosimetric calculations using DOSXYZnrc.

2.4.2.3 Component modules

One of its design features of BEAMnrc is that each part of the accelerator or source unit is considered to be a single component module which takes up a horizontal slab portion of the accelerator (Rogers et al 2010). Each CM can be considered as a building block of the radiotherapy accelerator. These component modules are re-usable and are all completely independent, communicating with the rest of the system in certain well specified ways. A diagram of an accelerator defined in BEAMnrc can be seen in Figure 2.17.

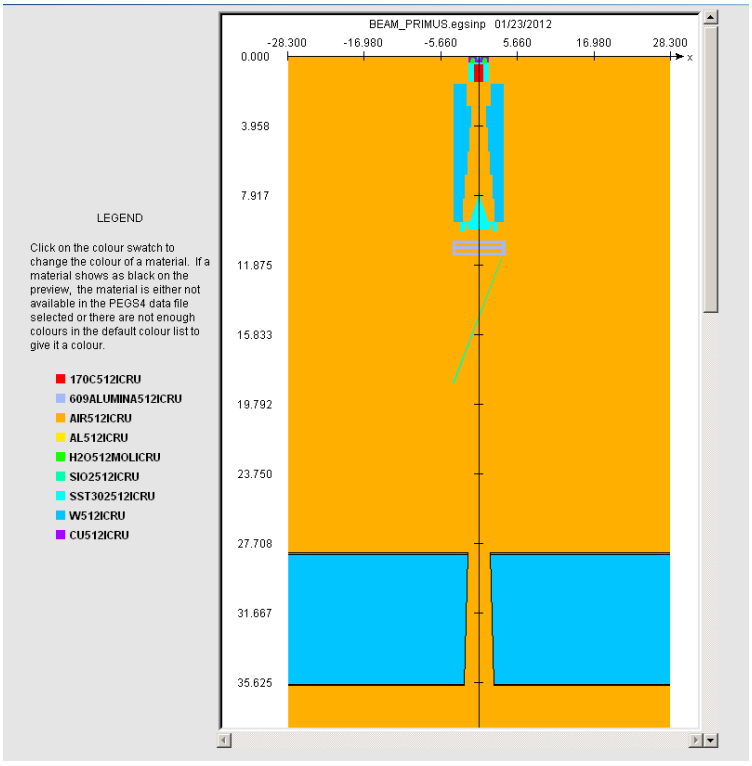

Figure 2.17: Diagram in the XZ plane of a Siemens PRIMUS accelerator created using the BEAMnrc GUI. The target, primary collimator, flattening filter and secondary collimator are easily distinguished in the image. Each of them is defined using the appropriate CM and the gaps between them are filled with air by the BEAMnrc main routine.

Each CM is made based on two MORTRAN source files. The MORTRAN macros specific to the component module CMNAME are contained in CMNAME_macros.mortran. These macros are used by BEAMnrc, the EGS subroutines and subroutines of the CM. The set of subroutines that define CMNAME are contained in CMNAME_cm.mortran. These subroutines are:

- HOWFAR_$CMNAME: it is used during the simulation to define the geometry of the component module (boundaries) and setting region-dependent parameters.
- HOWNEAR_$CMNAME: it calculates the perpendicular distance to the nearest boundary (i.e. not along the particle trajectory).
- WHERE_AM_I_$CMNAME: it is used to determine the region where the particle is upon entry into component module CMNAME.
- INPUT_$CMNAME: it manages the input from either the interactive user or the parameter definition file (i.e. .egsinp file) for information related to component module CMNAME.
- ISUMRY_$CMNAME: it writes the summary of the inputs for the component module.

2.5 ModuLeaf mini MLC

2.5.1 Mechanical and geometrical properties

The ModuLeaf mini MLC consists of 80 leaves with a maximum field size of 12x10 cm^2, with a maximum over travel of 5.5 cm, and a leaf width of 2.5 mm at isocenter. There are two different kinds of leaves, labelled “upper” and “lower”, where the “lower” leaves are shifted 0.55 cm towards the isocenter. Focusing in the Y direction is achieved by a convergent cross-section of each leaf and a quasi-focusing effect in X direction is achieved by the shape of the front edge of each leaf exhibiting two angles. Cross-sectional views of the leaves are shown in Figure 2.18. Rather than using a tongue-and groove design to avoid inter-leaf leakage, all leaves are tilted 0.76 ° counter clockwise in such a way that the convergent leaf sides are not exactly focusing at the nominal focus on the beam central axis but around 1.9 cm laterally apart. Further details of the mini MLC are listed in Table 2.1.

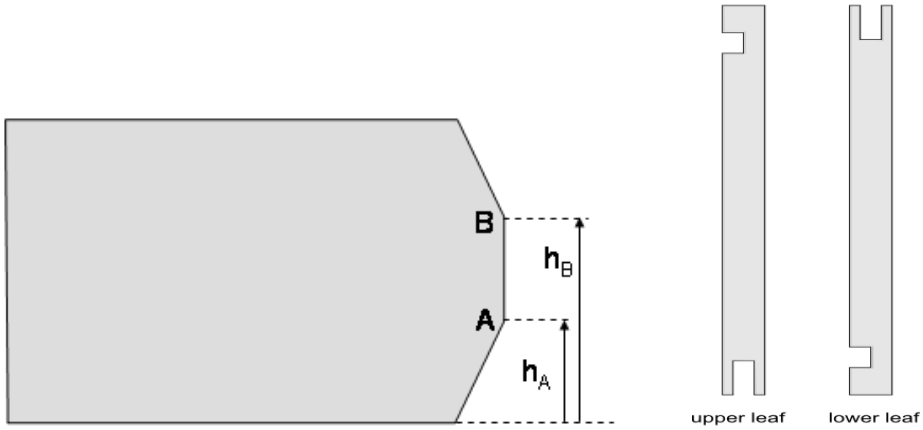

Figure 2.18: Left: Schematic drawing of the angled, quasi-focusing leaf edge design consisting of three straight lines; h_A and h_B denote the distance of point A and B from the collimator (bottom) plane. Right: Basic cross sectional view of the leaves in the ModuLeaf mini MLC. There are two kinds of leaves named “upper” and “lower”. The latter are shifted 0.55 cm towards the isocenter of the accelerator while the points A and B remain at the same distance from the source.

Table 2.1: Geometrical and mechanical properties of the ModuLeaf mini MLC

Item	Value
Number of leaves	40
Leaf width at isocenter	2.5 mm
Leaf height	7 cm
Maximum field size at isocenter	12 cm x 10 cm
Source to collimator distance (distal side) (as mounted at Siemens PRIMUS)	67.86 cm
Maximum leaf travel over central axis	5.5 cm
Leaf focusing design	Angled leaf edge in X, focusing in Y
Positioning tolerance at isocenter	0.5 mm
Leaf velocity at isocenter	2 cm s^{-1}
Weight	43 kg
Tilt angle	0.76 °

Calculation of field size specific parameters requires to establish the relation between leaf positioning at source-collimator distance and the associated field size at isocenter distance. With the leaf edge design as shown in Figure 2.18 left, the field edge at isocenter distance is defined by the position of point A or B depending on whether leaves X1 or X2 are in open or in overtravel position. Table 2.2 summarizes the calculation rules in order to obtain the position of the front plane at the collimator plane, $X_{\text{collimator plane}}$ when the field edge X is to be obtained at the isocenter plane.

Table 2.2: Calculation rules to obtain the actual position of a leaf front edge, $X_{collimatot\ plane,}$ at source collimator distance SCD from a given field edge position X at isocenter distance SID. h_A and h_B refer to Figure 1.

	$X_{collimator\ plane}$	
	X1 leaf	X2 leaf
Open position	$-X \cdot \frac{SCD-h_A}{SID}$	$X \cdot \frac{SCD-h_A}{SID}$
Overtravel position	$-X \cdot \frac{SCD-h_B}{SID}$	$X \cdot \frac{SCD-h_B}{SID}$

Field edges and field sizes are subsequently given as those at isocenter distances. The actual leaf positions are always modelled using the rules as given in Table 2.2.

2.5.2 Monte Carlo modelling

An existing DYNVMLC component module, originally developed to model the Varian Millennium 120 leaf collimator (Heath and Seuntjens 2003), was used as a starting point and then modified to generate the new component module, called MMLC. It now models in particular the details of the specific leaf design of the ModuLeaf mini MLC. The leaf definition scheme of sub-regions and the divergence in the direction perpendicular to the movement of the leaves were maintained. The shape of the leaf ending was changed in order to model the quasi focusing effect in the direction of movement and the leaf bank tilting was introduced. The INPUT subroutine was accordingly modified.

The geometry of the two leaf types, upper and lower, is illustrated in Figure 2.19. In the INPUT subroutine the dimensions of the leaves are entered on a leaf-by-leaf basis and stored in an array using the leaf number as an index. Based on the leaf number parity (i.e. whether the number is even or odd) the leaf type is determined. A series of geometry checks are performed to test for overlap between adjacent leaves and any other errors in the leaf geometry.

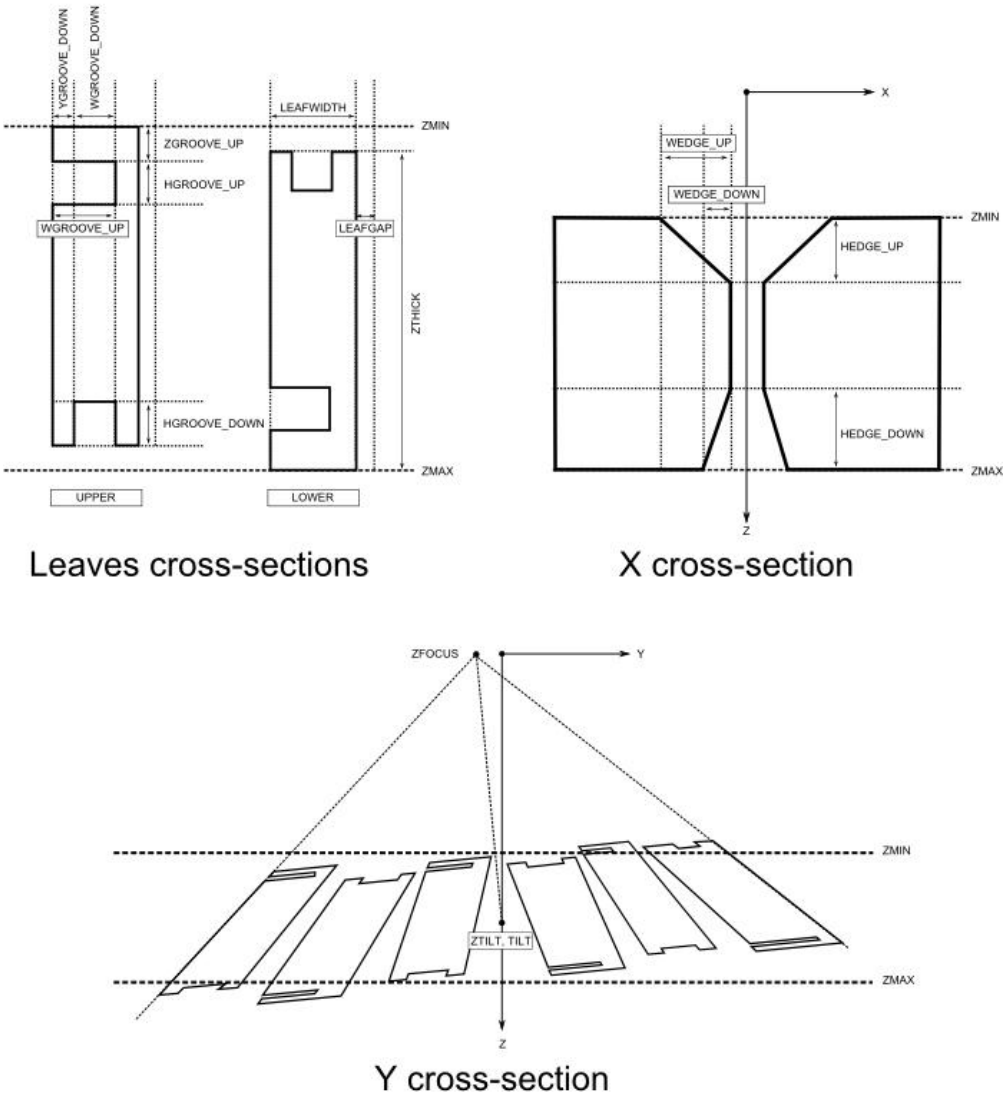

Figure 2.19: Cross-sections of the MMLC CM with some input values. On the right (Y cross-section) can be appreciated the tilt of the leaves in an angle TILT with respect to the point ZTILT located in the central axis.

Each leaf is divided into a set of sub-regions bound by a series of planes in the X, Y and Z directions. Depending on the orientation of the leaves the X or Y planes diverge to account for the focused leaf sides and the Z planes are rotated accordingly to the focused arrangement of the leaves. Finally the entire set of X or Y planes and Z planes are rotated together to account for the tilting of the entire bank of leaves. The quasi-focused leaf edge design is modelled with a set of three planes. The subroutine HOWFAR determines the intersection of the current trajectory of the particle with a plane defining the leaf medium boundary and the subroutine HOWNEAR determines the distance between the particle to the nearest boundary.

A major modification is that the gaps between the leaves are modelled as additional sub-regions. The reason was that we found a problem in the modeling of leaf gaps in the existing DYNVMLC CM. The problem is that photons which happened to be transported into a leaf gap were not correctly further transported as illustrated in Figure 2.20. In our new CM the sub-regions of the leaves and the gaps are treated identically with the difference that the medium of each sub-region is now defined to be either tungsten or air. All leaf dimensions

where obtained from the manufacturer's technical drawings. Tungsten alloy with a density of 19.3 g/cm^3 was used for the leaf material.

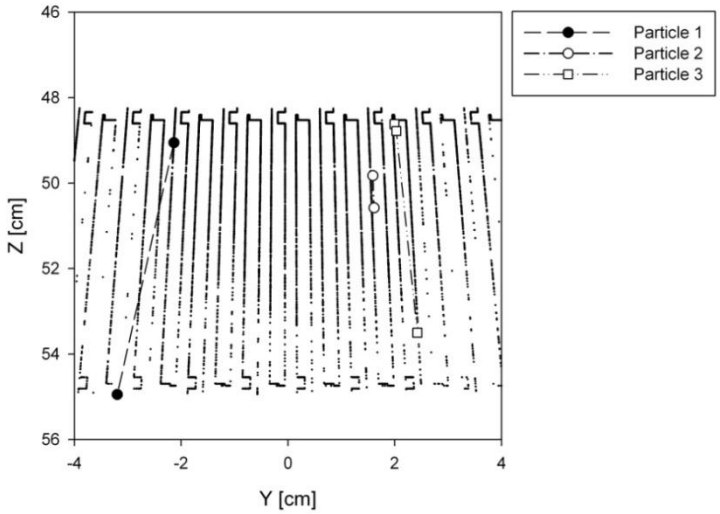

Figure 2.20: Transportation of particles through the DYNVMLC CM. The inputs where chosen in order to reassemble the ModuLeaf mini MLC geometry. Particle 1, originally in the gap between two leafs, is immediately transported to the bottom of the CM, regardless the presence of leaves in the fly path of the particle. Particles 2 and 3 are correctly transported until they reach the border of the leaf.

2.5.3 Validation of MMLC component module

2.5.3.1 Validation of the component module geometry by ray tracing

To test the geometry of the new CM, the ray tracing method (Heath and Seutjens 2003) was used. Macros in the CM code allow a particle's position to be scored whenever it crosses a boundary between the leaf material and air. The points can be plotted either as 2D or 3D projections to visualize the leaf geometry. A 12 x 10 cm^2 field incident on the component module was used to obtain the ray tracing images.

2.5.3.2 Comparison of measurements and calculated dosimetrical parameters

The following parameters have been used for the comparison:

- output factors
- dose profiles
- field size and field edge position
- shape of penumbra
- MLC leakage and transmission

2.5.3.3 Output factors

Output factors measurements were performed using an ionization chamber (Type 31002, PTW, Freiburg) at 5 cm depth in a water phantom at 100 cm SSD. The reference field is 10 x 10 cm^2 at same depth and SSD and the results were obtained at field settings from 0.5 x 0.5 cm^2 to 10 x 10 cm^2. The accelerator jaws settings were kept constant at 12 x 10 cm^2. Output factors were also obtained using the DOSXYZnrc user code and compared with

the output factors obtained from the measurements. A voxel size of 0.35 x 0.35 x 0.5 cm^3 was used in a water phantom to score absorbed dose at the point of interest, centered with respect to the beam axis at a depth of 5 cm. The absorbed dose is then normalized with respect to the dose scored in the monitor chamber of the PRIMUS, in order to take into account the collimator exchange effect in the accelerator head, and with respect to the dose obtained at the 10 x 10 cm^2 reference field.

2.5.3.4 Lateral dose profiles

The coordinate system and the notation used to denote the position of the leaves follow the recommendations given in IEC 61217 (1996).

A series of X profiles were obtained from various rectangular fields shaped with the MMLC. For each rectangular field, the border lines in Y were set constantly to +4 cm and -4 cm. The border lines in X were set according to the following subsets: (a) X_1= -5.5 cm, X_2= -5, -4, -3,…, +5 cm; (b) X_1= -5, -4, -3,…, +5 cm, X_2= +5.5 cm.

Measurements of dose distributions were performed using radiographic film (Kodak X-Omat-V) and digitalized using a film scanner (FIPS Plus, PTW, Freiburg, Germany). The files obtained were analyzed using the Mephisto mc^2 analysis software (v.1.8, PTW-Freiburg, Germany). For each rectangular field a series of X dose profiles were obtained, separated by a distance of 5 mm along the X axis. After the different profiles were obtained (usually 4 or 6 profiles per field) they were averaged in order to improve the quality of the measurement. Finally, the profile obtained is normalized with respect to the nominal central point of the field. Off-axis profiles were calculated in a water phantom using the DOSXYZnrc user code and compared with the profiles obtained from the measurements. For each field an off-axis profile in X direction was scored and then normalized in the same way as the measurements. A row of voxels of 0.1 x 2.0 x 5.0 cm^3 centered with respect to the field axis and at a depth of 5 cm was used to score the dose profile.

2.5.3.5 Field size and field edge position

The values used to calculate the field edge and the field penumbra were the position of the 50% isodose and the distance between the 20% to the 80% isodose line, respectively. Calculated and measured results of field edge positions and field sizes were compared for a number of collimator settings.

2.5.3.6 MLC leakage and transmission

For measurement of the MLC interleaf leakage a film was exposed to 1000 MU with the accelerator jaws settings kept at 12 x 10 cm^2 and the MLC leaves blocking the field. A lateral dose profile was obtained at the isocenter in Y direction. Finally, the profile obtained was normalized with respect to the dose of the open field. This dose was obtained using an ionization chamber (Type 31002, PTW, Freiburg) at 5 cm depth in a water phantom at 95 cm SSD for a 10 x 10 cm^2 open field. MLC interleaf leakage was also calculated in a water phantom using DOSXYZnrc user code and compared with the measurements. A row of voxels of 2.0x0.1x5.0 cm^3 centered with respect to the field axis and at a depth of 5 cm was used to score the interleaf leakage profile and then a single voxel was used to obtain the reference dose at the center of the open field. Finally the profile was normalized in the same way as the measurements.

2.6 Additional developments

Currently the DKFZ platform is being used for the simulation of two different collimators, the IRIS collimators mounted on the Accuray Cyberknife linear accelerator and a new compact MLC prototype developed at the DKFZ.

2.6.1 IRIS Collimator

Traditionally, fixed divergent circular collimators have been employed to shape x-ray beams used for radiosurgery. Such collimators have very low collimator transmission, sharp penumbrae and perfect field size reproducibility. These characteristics are important because large-dose single-fraction or hypofractionated treatments require submillimetric geometric beam delivery accuracy and steep dose gradients in order to spare adjacent organs at risk (Echner et al 2009). The CyberKnife Robotic Radiosurgery System (Accuray Incorporated, Sunnyvale, CA, USA) is designed to deliver radiation to targets located anywhere in the body. This system consists of a 6MV X-band LINAC mounted on a robotic manipulator that is aligned continually throughout each treatment fraction by a stereoscopic x-ray image guidance system. Each treatment beam is shaped using a fixed circular collimator selected from a set of 12 that produce beam diameters ranging from 5 to 60 mm at a nominal treatment distance of 80 cm from the therapeutic x-ray target (Kilby et al 2009).

The purpose of an iris collimator that allows the field size to be varied during treatment delivery is to enable the benefits of multiple field size treatments to be realized with no increase in treatment time due to collimator exchange or multiple exposures of the robotic manipulator by allowing each beam to be delivered with any desired field size during a single exposure. The IRIS variable aperture collimator (Accuray) incorporates twelve 60 mm tall prism-shaped tungsten-copper allow segments in two banks of six for shaping the radiation field (Echner et al 2009). A diagram of the IRIS collimator is shown in Figure 2.21.

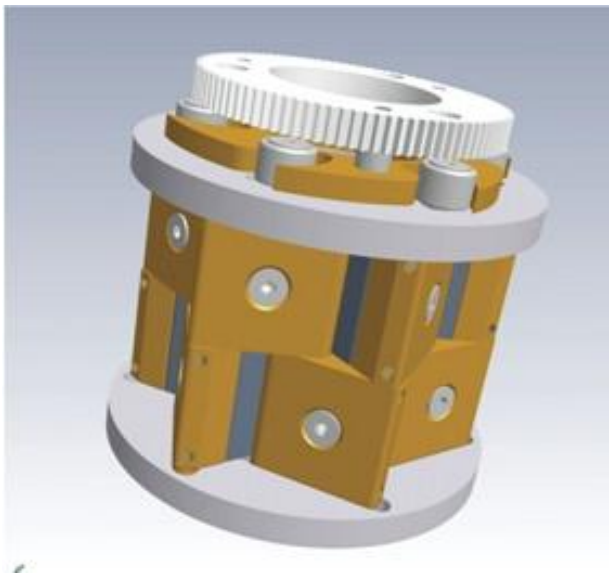

Figure 2.21: The IRIS variable aperture collimator (Echner et al 2009).

The DKFZ platform has been used to develop a CM for the IRIS collimator. The CM simulates one bank of six segments; therefore the CM must be attached two times in order to model the entire IRIS collimator system. The main inputs are the segments length, the size of the aperture and the rotation of the leaf bank with respect to the beam central axis. Based on the aperture size the code models the translation of the segments into and out of the beam.

In order to test the component module for the IRIS collimator the ray tracing method was used together with the MC calculation of output factors, dose profiles and depth dose curves using the DOSXYZnrc user code in a water phantom at 80 cm SSD. For the calculation of output factors a voxel size of 0.35 x 0.35 x 0.5 cm^3 was used in a water phantom to score absorbed dose at the point of interest, centered with respect to the beam axis at a depth of 5 cm. A row of voxels of 0.1 x 2.0 x 5.0 cm^3 centered with respect to the field axis and at a depth of 5 cm was used to score the dose profiles and the depth dose curves were scored using a row of 0.5 x 0.5 x 0.2 cm^3 along the beam central axis.

2.6.2 Compact MLC prototype

A new compact MLC prototype is being developed at the DKFZ for the Accuray Cyberknife LINAC. In order to study the properties of this prototype a new CM named CMLC was created. The compact MLC has 41 pair of leaves of the same kind, and as in the ModuLeaf MLC, rather than using a tongue-and groove design to avoid inter-leaf leakage, all leaves are tilted counter clockwise in such a way that the convergent leaf sides are not exactly focusing at the nominal focus on the beam central axis.

At a first step two leaf edge designs were tested, the first design corresponds to the shape of the front edge of each leaf exhibiting three sides, as in the ModuLeaf mini MLC, and the second using five sides. Therefore the first design is labelled “three-sided” and the second “five-sided. Both designs are shown in Figure 2.22.

The objective of this MC study is to determine which design offers the smallest penumbra across the full range of leaf motion and lowest leakage dose at the leaf meeting point, for multiple leaf tip-tip gaps and off axis distances. As a first step the correctness of the simulated geometry is tested using the ray tracing method. Then the dosimetric calculations using MC were performed.

For conventional flattened beam profiles, the definition of the penumbra is based on the 20-80% dose values. This is not applicable in this case because the treatment head does not have a flattening filter. Therefore, the penumbra of the unflattened dose profile was derived from the spatial distance between the positions where the doses were 20 and 80% of the normalized dose $D_n = (D_u/D_f)D_{CAX}$ (Pönisch et al 2006). D_u is the dose at the inflection point of the penumbra region of the unflattened beam; D_f the dose at the inflection point of the flattened profile (with flattening filter) and D_{CAX} is the dose on the central axis of the flattened beam.

Unfortunately is not possible to obtain a flattened profile from the model of the Cyber-Knife treatment head; however it is possible to give estimation for the factor D_{CAX}/D_f. A value of $D_{CAX}/D_f = 2$ was estimated from previous dosimetric studies for similar MLCs. The inflection point in the penumbra regions was determined from the first derivative of the dose profiles.

The dose profiles were obtained using the DOSXYZnrc user code in a water phantom at 80 cm SSD with a centered row of voxels along X axis at 5.0 cm depth. In the case of the penumbra calculations the size of the voxels are 0.025 x 2.0 x 0.5 cm^3 in the penumbra region of the beam and 0.2 x 2.0 x 0.5 cm^3 elsewhere, allowing an appropriate spatial resolution of the dose profile, especially in the penumbra region. For the tip-tip leakage calculations a uniform voxel size of 0.1 x 2.0 x 0.5 cm^3 was used. For the calculation of the tip-tip leakages each profile were normalized with respect to the reference dose D_{ref} obtained from a 10 x 10 cm^2 field at the point of measurement placed in the beam central axis, 85 cm from the source and 5 cm depth in the water phantom.

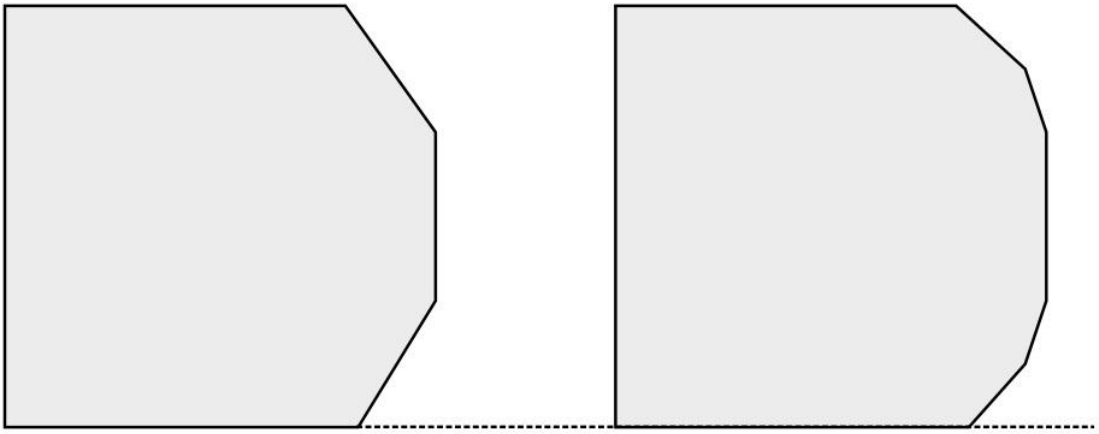

Figure 2.22: Leaf edge designs tested on the CMLC. Left: “three-sided”, right: “five-sided”.

3 RESULTS

3.1 The DKFZ Platform

3.1.1 Source file to build a BEAM code

A new MAKEFILE was created for the DKFZ platform that contains the complete list of the files needed to build the code for an arbitrary accelerator. The MAKEFILE used for the MC simulation of the ModuLeaf mini MLC is given in Appendix B.

3.1.2 Mortran code of the ModuLeaf MLC

The MMLC CM was created to simulate the ModuLeaf mini MLC, using the tools given by the DKFZ platform using the IVFC with its valuable debugging tool. With these tools the subroutines and macros used in the MORTRAN codes of the CM were carefully checked and tested. The MORTRAN source files MMLC_cm.mortran and MMLC_macros.mortran are given in Appendix C.

3.1.3 Data flow of the platform

Figure 3.1 shows the data flow when the beam code “beam_MMLC.f” is created and executed with the new CM. The CMs used to model the PRIMUS linear accelerator, with the ModuLeaf attached on it, are called by the MAKEFILE. The Mortran code of the MMLC CM is given in Appendix C.

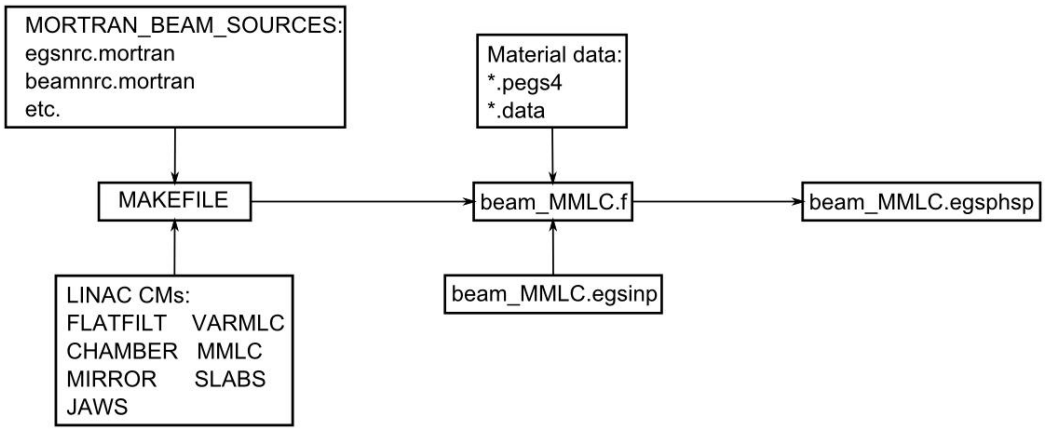

Figure 3.1: Data flow of the beam_MMLC.f BEAM code.

The phase space file beam_MMLC.egsphsp obtained after the MC simulation is then used as an input for the dose calculations using the DOSXYZnrc code.

3.2 Monte Carlo based characterization of the ModuLeaf mini MLC

3.2.1 Validation of CM geometry by ray tracing

Figure 3.2 shows cross sectional views of the MMLC of a group of leaves, plotting the y-z (a) and x-z (b) coordinates obtained from the ray tracing method. In the plotting of the YZ plane leaf all details such as the guiding grooves and the shift in height between the two different kinds of leaves are well visible. The air gaps between the leaves and the leaf bank tilting can

also be well observed. The front edge of the leaves showing the shape used to achieve the quasi-focusing effect in the X direction is reproduced in the XZ plane.

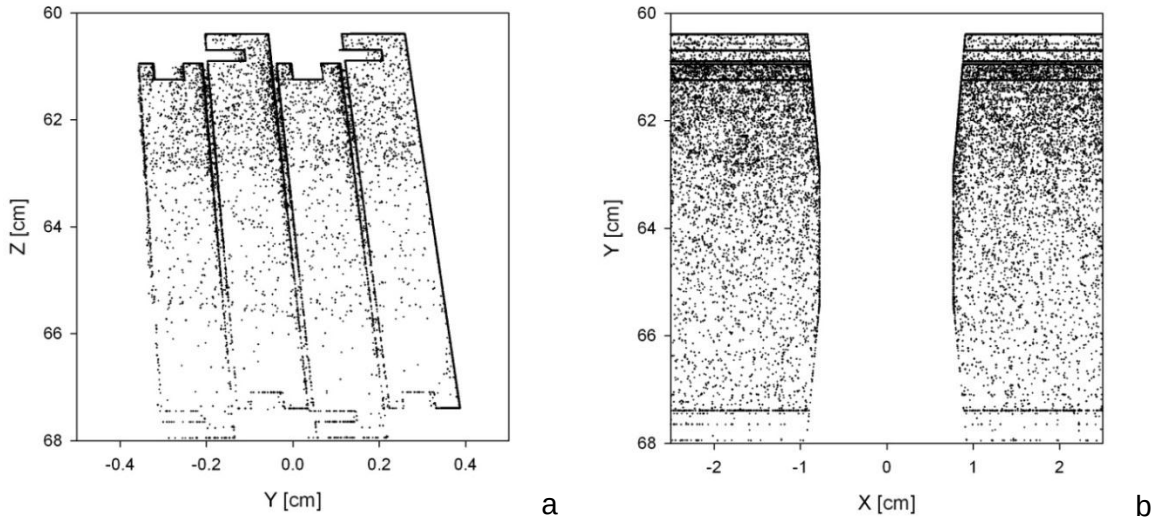

Figure 3.2: (a) YZ plane projection of a group of leaves obtained by ray tracing. The z axis represents the distance from the front face of the target. (b) XZ plane projection of the groups of leaves, the quasi-focusing effect shape of the front edge of the leaves is clearly recognizable. Note: the size scale is different in the figures.

Figure 3.3 shows the YZ plane projection of the entire leaf bank. Characteristics of the MLC like the number of leaves, the dimensions of the leaf bank and the position of the CM in the simulated geometry can be distinguished in (a). On the other hand, Figure 3.3 (b) corresponds to the upper part of the leaf bank. The position of the bank at the central axis is represented by the dashed line. The variation of the height of the leaves from the central axis to the sides of the MLC due to the focused arrangement of the leaves can be clearly appreciated.

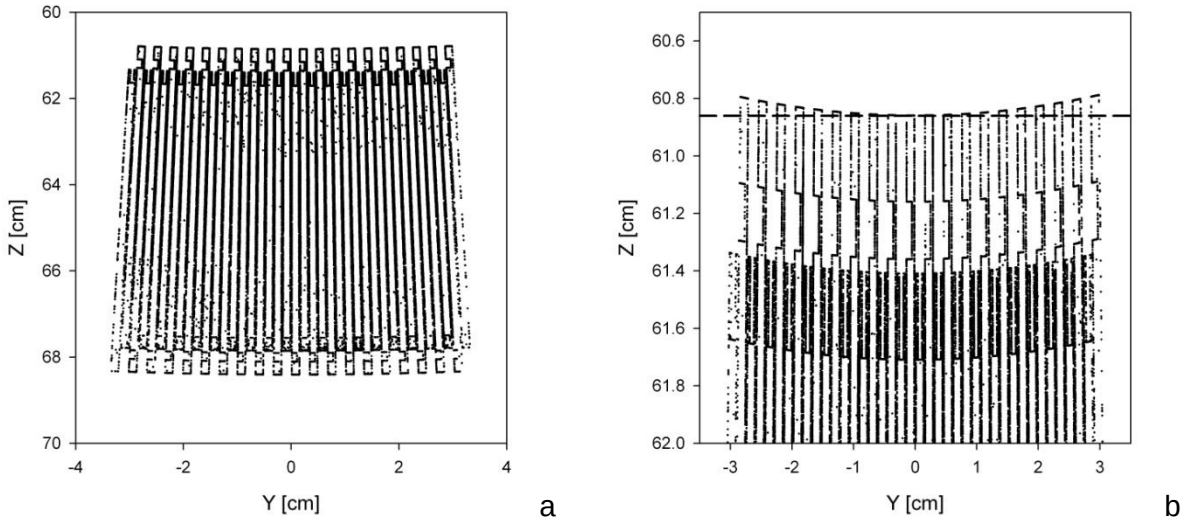

Figure 3.3: (a) YZ projection of the MMLC leaf bank. (b) Projection of the upper part of the leaf bank, the position of the MLC in the central axis is depicted by the dashed line.

3.2.2 Output factors

Figure 3.4 shows a comparison between the measured and MC calculated output factors. Both curves show a steep drop-off below 3 x 3 cm^2 field, which is expected due to the shielding by the collimator of the scattered photons originating from the flattening filter, hereby creating an effective extended source. The uncertainty associated with the MC calculated output factors is 0.5%.

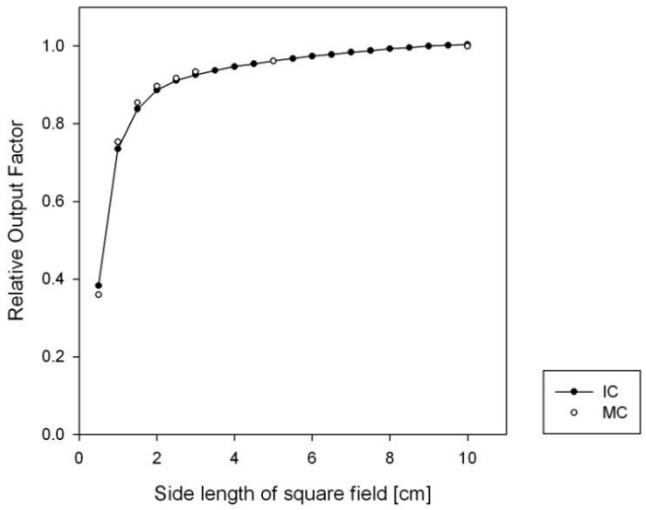

Figure 3.4: Comparison of measured and MC calculated OFs.

3.2.3 Penumbra and field edge definition

The problem of the relation of the field size at isocenter and the mechanical position of leaves with "rounded" edges has been analyzed by several authors (LoSasso et al 1998, Graves et al 2001). The problem is that the transmission at the field edge may vary depending on the position of a leaf due to its rounded shape. The field size at isocenter therefore might not entirely only dependent on the leaf position.

Figure 3.5 shows a summary of the dose profiles in the X direction, obtained from the dose measurements and the MC simulations. The uncertainty associated with the MC simulations is 1.0%.

The position of a leaf in the leaf bank is described depending on its location with respect to the beam central axis. When the leaf edge crosses the beam central axis the leaf is in an over-travel position, otherwise the leaf is in an open-position.

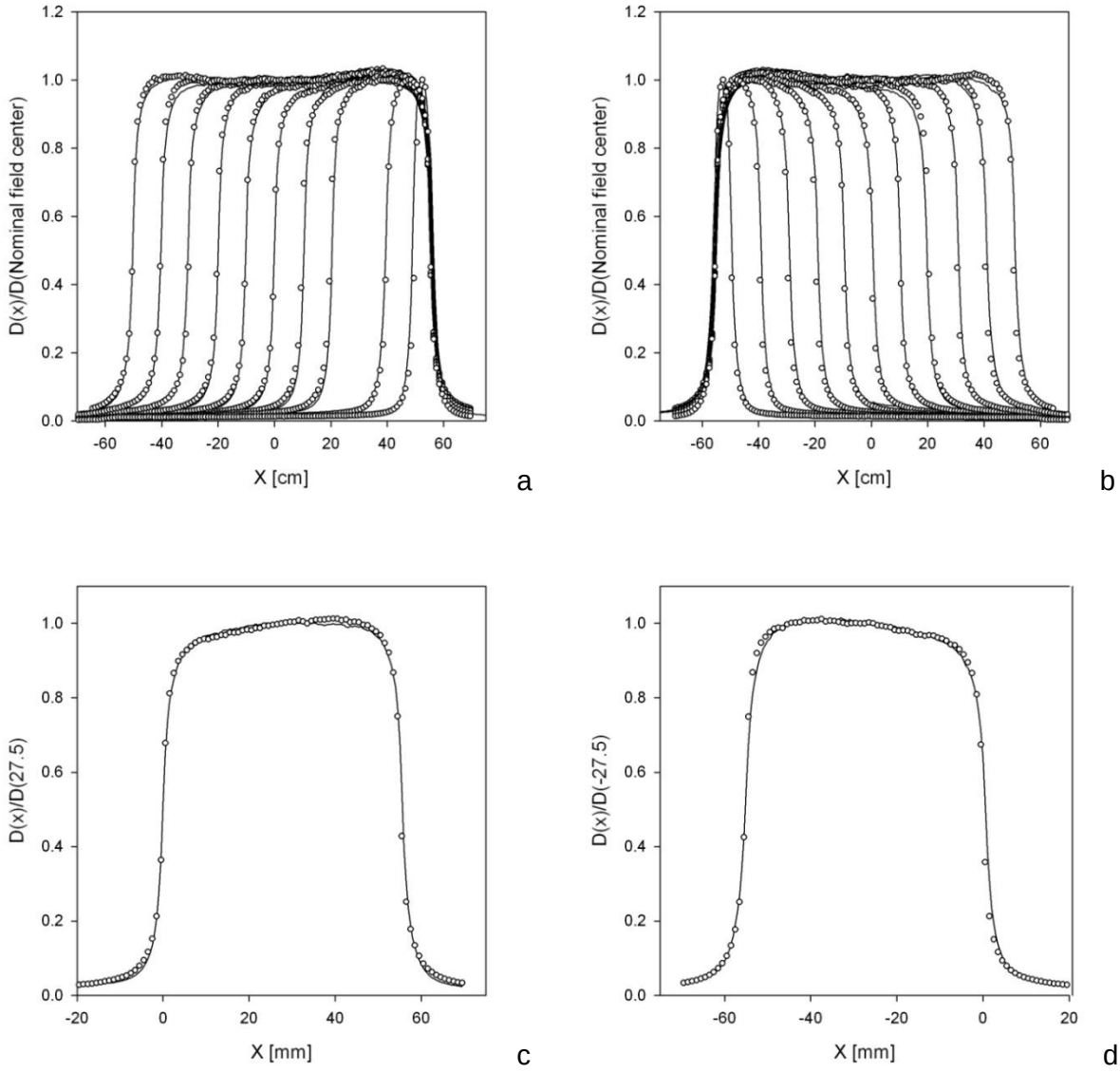

Figure 3.5: Comparison of dose profiles in the direction of movement of the leaves. The solid lines indicate dose profiles obtained with the radiographic films and the dots indicate BEAMnrc/DOSXYZnrc simulations. (a) Series of dose profiles varying X1 leaves from -5.0 cm (left) to 0.0 cm while maintaining X2 leaves at +5.5 cm. (b) Series of dose profiles varying X2 leaves from +5.0 (right) to 0.0 cm while maintaining X1 leaves at -5.5 cm. (c) Dose profile with X1 leaves at 0.0 cm and X2 leaves at +5.5 cm. (d) Dose profile with X2 leaves at 0.0 cm and X1 leaves at -5.5 cm.

Figure 3.6 shows an evaluation of the field size accuracy for open and over-travel positions of the leaves. Evaluation was performed by determining the difference between the nominal field size and that obtained from the measurements with dosimetric films or from MC simulations as a function of nominal leaf position. The absolute value of the leaf positions was used, so that it is possible to compare negative and positive positions of the leaf bank.

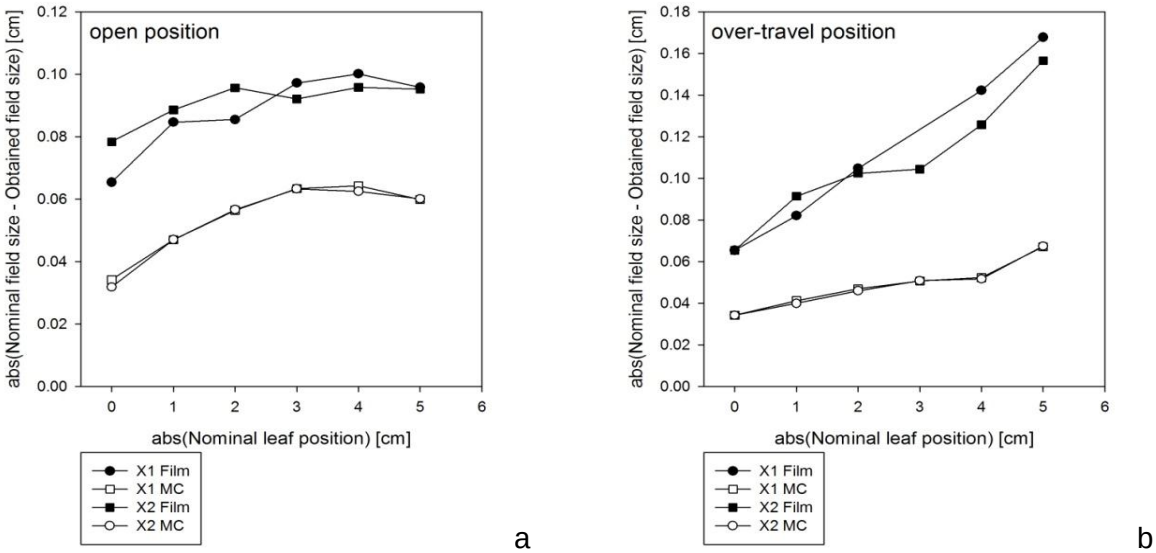

Figure 3.6: Evaluation of the accuracy of field edge positioning. Comparison of obtained field size (50% isodose line distance) with respect to the nominal leaf position for Film and MC simulations with the leaves in (a) open position and (b) over-travel position.

Figure 3.7 shows an evaluation of the actual position of the leaves edge. The difference between the nominal leaf edge position and that obtained from the MC simulations was determined as a function of nominal leaf position.

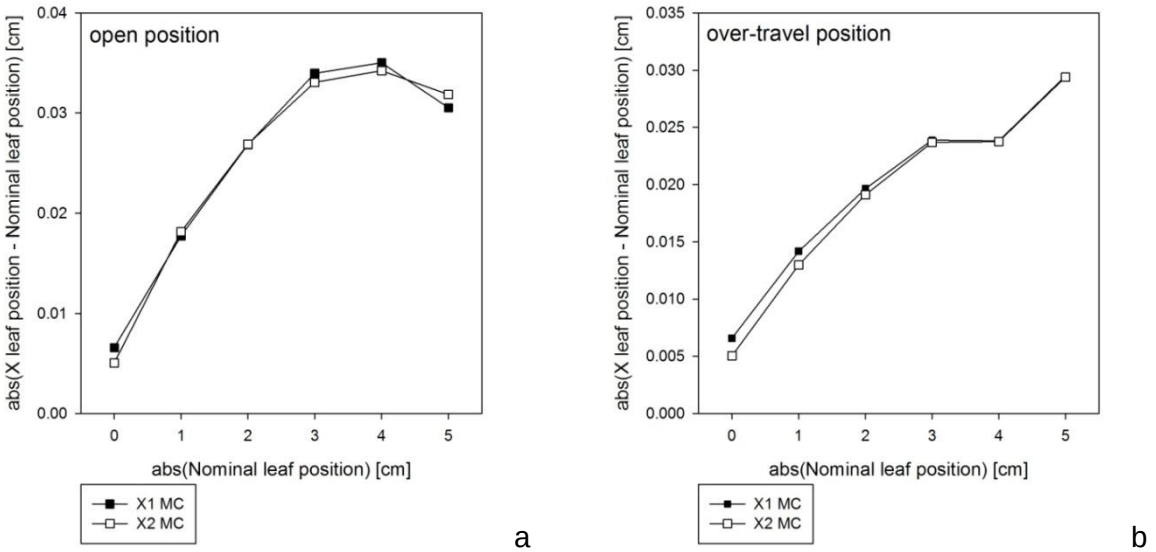

Figure 3.7: Comparison of the obtained leaf position from MC simulations (position at 50% relative dose) with leaves in (a) open position and (b) over travel position.

3.2.4 MLC leakage and transmission

Results of leakage and transmission are shown in Figure 3.8. The mean uncertainty of the MC calculations is around 1.0%.

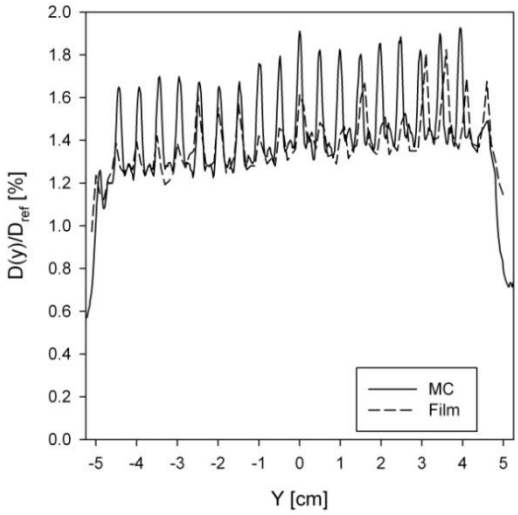

Figure 3.8: Comparison of transmission and interleaf leakage profile for the ModuLeaf MLC. The dashed line indicates the profile obtained using radiographic film and the solid line corresponds to MC simulations.

3.3 IRIS Collimator

Figure 3.9 shows a front view of the IRIS CM using the ray-tracing method. The prism-shaped segments can be easily distinguished together with the characteristic aperture shape of this collimator. Figure 3.10 shows relative output factors and Figure 3.11 presents off-axis dose profiles and depth dose curves obtained for several field sizes. The uncertainty associated with the MC simulations is 0.5%.

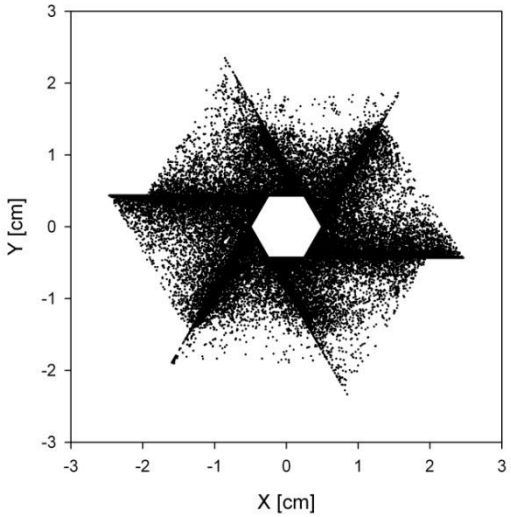

Figure 3.9: XY projection of the IRIS CM obtained by ray-tracing.

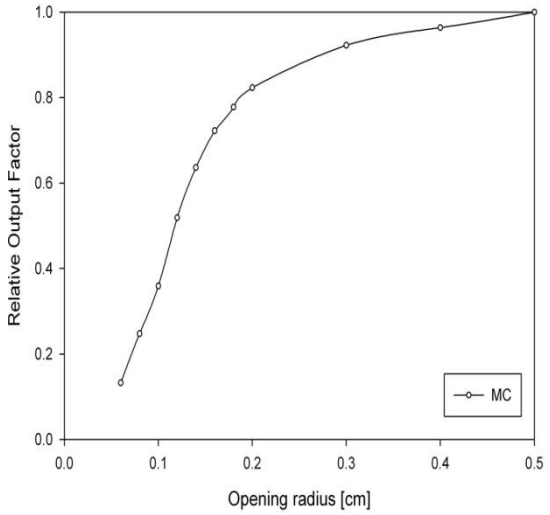

Figure 3.10: MC calculated OFs for the IRIS CM.

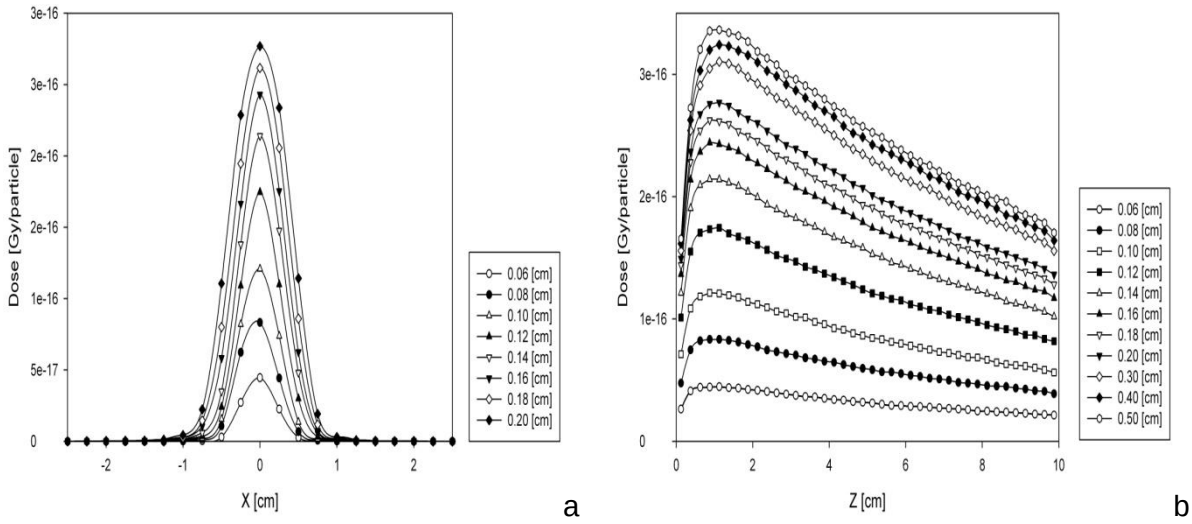

Figure 3.11: MC calculated off-axis profiles and depth dose curves for the IRIS CM.

3.4 Leaf edge design study of the Compact MLC prototype

In Figures 3.12 and 3.13 cross-sectional views of the CM obtained by ray tracing are shown. The front edge designs can be recognized in the XY in the plotting of the XZ plane. In the YZ plane can be seen the leaf details as the guiding tips in the extremes of the leaves.

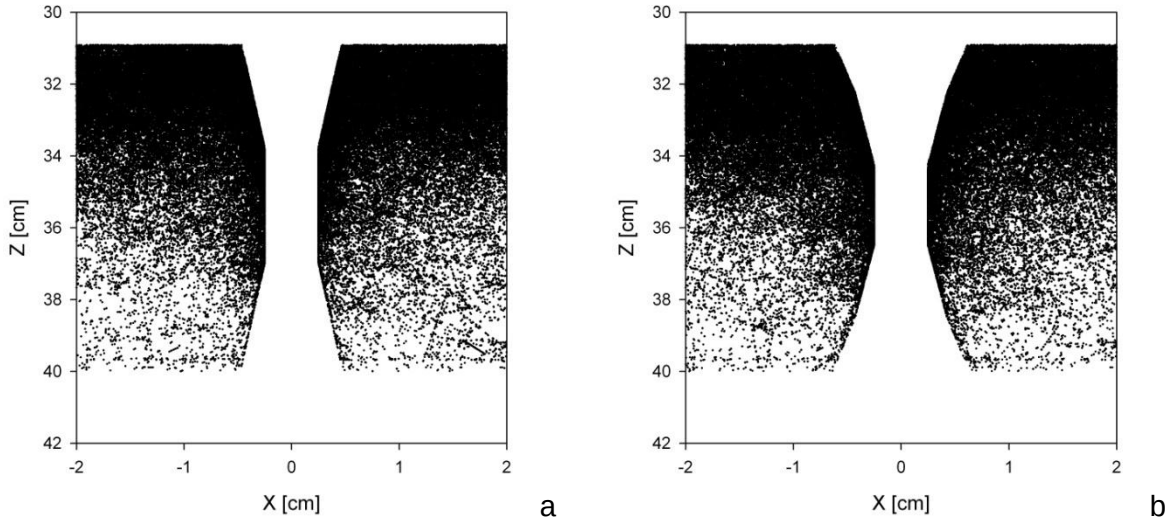

Figure 3.12: XZ cross-sectional views of the CMLC CM obtained by ray-tracing. (a) Three-sided and (b) Five-sided designs.

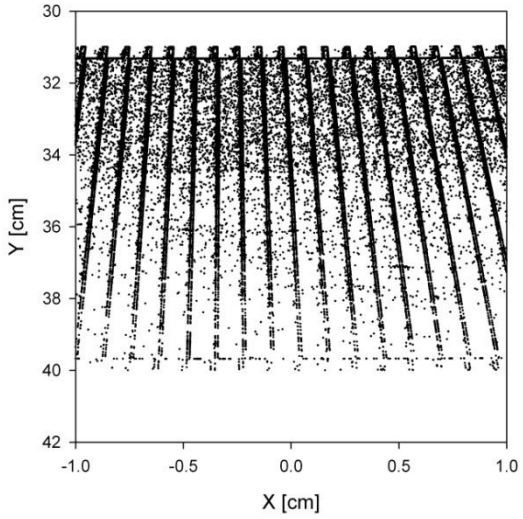

Figure 3.13: YZ cross-sectional view of the CMLC CM obtained by ray-tracing.

The results for the penumbra calculations are shown in Table 3.1. In all the cases the penumbra for the “three-sided” design is lower that the “five-sided” design. In particular the penumbra for the first design ranges from 4.36 mm for the first field (6 cm symmetrical field centred with respect to the beam central axis) to 3.00 mm for the last field (4 mm field at 2.8 cm of the beam central axis). It can be seen that the penumbra decreases with the field size.

Table 3.1: Penumbra calculations of the prototype MLC for the three-sided (3s) and five-sided (5s) designs.

			Mean Penumbra [mm]		Δ (5s-3s) [mm]
# Field	X1 [cm]	X2 [cm]	3s	5s	
1	-3.0	3.0	4.36	5.26	0.89
2	-1.5	1.5	4.19	4.36	0.17
3	-3.0	0.0	3.83	4.43	0.60
4	-0.2	0.2	3.60	3.75	0.15
5	-3.0	-2.6	3.00	3.75	0.75

The results for the dose leakage between opposing leaves are summarized in Table 3.2. The dose profiles are shown in Figure 3.14.

Table 3.2. Dose leakage calculations between opposing leaves.

			Relative Maximal Dose [%]		Δ(5s-3s)
# Field	Off axis distance [cm]	GAP [mm]	3s	5s	
6	1.5	0.0	0.87	2.58	1.71
7	1.5	0.2	1.12	3.82	2.70
8	1.5	0.4	1.59	5.95	4.36
9	3.0	0.0	0.46	0.58	0.12
10	3.0	0.2	0.48	0.60	0.12
11	3.0	0.4	0.51	0.62	0.11

For each dose profile the relative maximal dose (i.e. the maximal dose divided by the reference dose D_{ref}) and the difference between the leaf edge designs are shown. In all fields the dose leakage is lower for the three sides design. At 1.5 cm of the central axis the difference between the leaf edge designs increases with respect to the gap between the opposing leaves. At 3.0 cm this difference keeps approximately constant.

The results obtained allowed us to conclude that the “three-sided” design offers the lowest penumbra and tip-tip dose leakage values. Hence this design was chosen for the compact MLC prototype. The uncertainty of the MC calculations is around 1%.

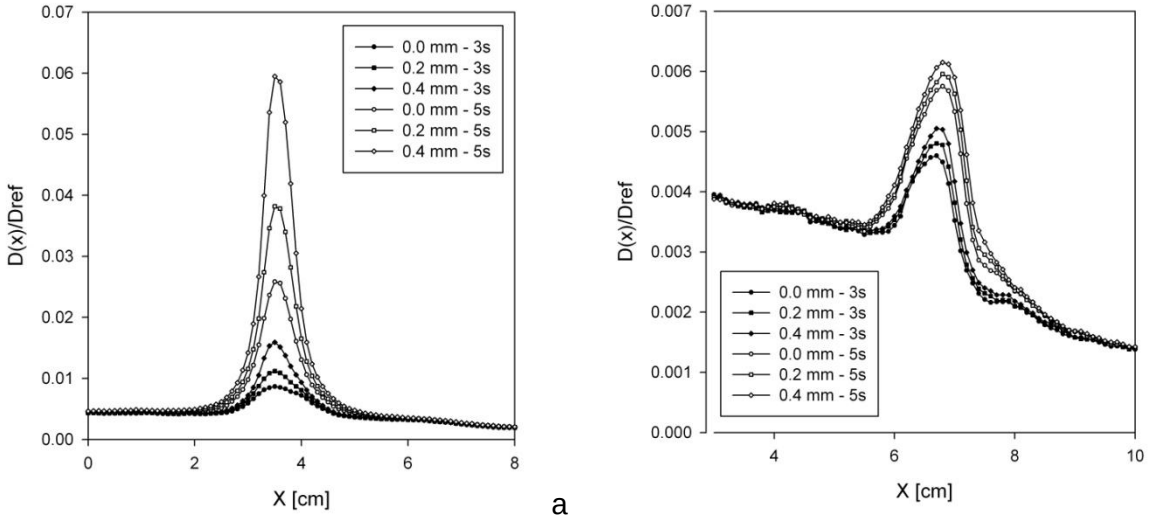

Figure 3.14: Dose leakage between opposing leaf edges at (a) $1.5\ cm$ of the beam central axis in the collimator plane and (b) $3.0\ cm$ of the beam central axis.

4 DISCUSSION

4.1 The DKFZ Platform

The DKFZ platform was created to allow an easier implementation of the EGSnrc Monte Carlo code system to run with the IVFC under Windows. The simplified structure of our platform, when compared with the original system, gives us an improved and effective way to realize our ideas in the field of Monte Carlo modelling. This is considered as an essential element to utilize the potential of Monte Carlo method in the field of medical physics.

In particular the debugging system of the IFVC was found to be an important development tool. It allows a better analysis of the execution process and facilitates the detection of errors in the code. It also facilitates the writing and development of the code, especially for a person that had a little background in the field of programming, as it was the case of me. The use of the IVFC with its valuable debugging tool is therefore considered as a very important contribution for the learning curve of the FORTRAN language.

For example, the different calculations involved in the transport of a single particle can be tested in order to assure the correctness of the modelling of the CM. Moreover, if a certain case is needed (e.g. a particle in the interleaf gap region) then a break in the code can be inserted in order to stop the execution flow when the wanted requirement is fulfilled. Then the execution of the program can be resumed to study the following transport of the particle. This technique was found to be extremely helpful to study and finally analyse the wrong particle transport in the interleaf gap region in the DYNVMLC CM as described in the section 2.5.2.

4.2 Simulation of the ModuLeaf mini MLC

4.2.1 The ray tracing method

The ray-tracing method allows a direct visualization of the simulated geometry and it gives a first proof of the correct modelling of the CM. Physical properties of the ModuLeaf could be checked against the technical drawings and data available.

The CM is divided in regions whose boundaries are defined analytically by planes. Therefore the distances calculated by HOWFAR and HOWNEAR subroutines are obtained using analytical geometry. Moreover, the focused arrangement of the leaves and the tilting of the leaf bank imply a rotation of these planes with respect to the focus point and tilting point, respectively. The correctness of these operations can be easily checked using the ray-tracing method. The rotation of a point in the Cartesian plane is executed using the rotation matrix R:

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{58}$$

that rotates a point in the XY- Cartesian plane counter clockwise through an angle θ about the origin (see Figure 4.1). The last sentence is true if the coordinate system is right handed (e.g. if the positive X axis points in the right direction, then the Y axis points upwards). Therefore if the Y axis points downwards, the rotation matrix R would perform a clockwise rotation and R must be adequately changed.

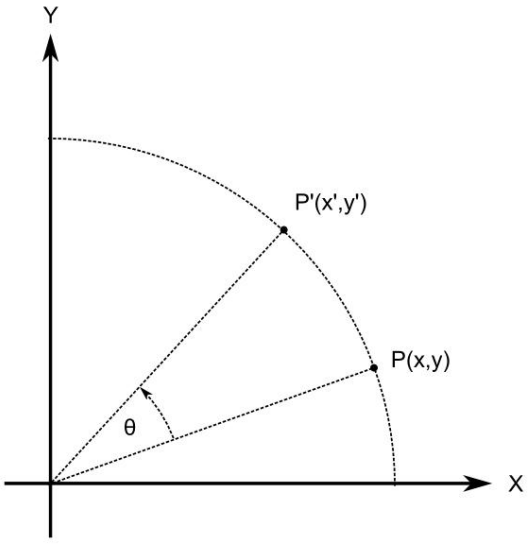

Figure 4.1: Counter clockwise rotation of the point $P = (x, y)$ with respect to the origin in an angle θ. The resulting point is $P' = (x', y') = R(\theta) \cdot (x, y)$.

In the simulated geometry the Z axis points downwards, hence a counter clockwise rotation through an angle θ of a point in the YZ plane is performed using the rotation matrix R':

$$R'(\theta) = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{59}$$

If the rotation matrix R is used instead of R' in the focused arrangement of the leaves, the ray-tracing image in Figure 4.2 is obtained. As can be seen, instead of a concave front surface the leaf bank presents a convex shape, i.e. the central leaves are closer to the focus point than the leaves at the sides of the leaf bank. Using the ray-tracing image this mistake can be easily detected.

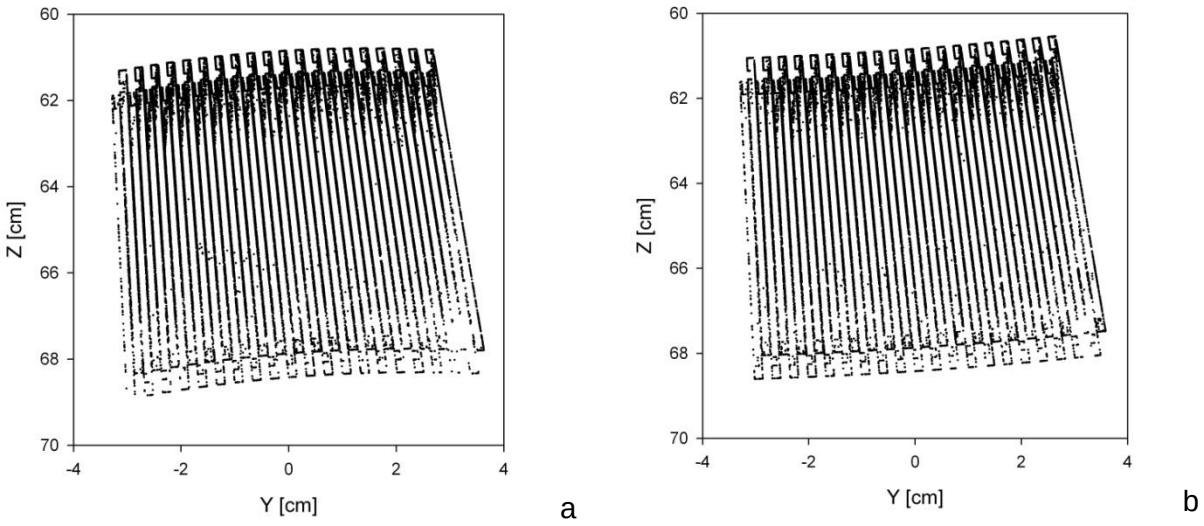

Figure 4.2: Focused arrangement of the leaves. (a) Incorrect focused arrangement of the leaves, when the rotation matrix R is used. The front surface of the leaf bank becomes convex. (b) Correct focused arrangement of the leaves, when the rotation matrix R' is used.

4.2.2 Comparison with the dosimetric characterization of the ModuLeaf

Although the ray-tracing method allowed us to test the correct simulation of the geometry it does not assure that the MMLC CM describes properly the dosimetric properties of the ModuLeaf. As an example, from Figure 2.20 it was not possible to detect the interleaf gap transport error in the DYNVMLC, because the particles that fall in the gaps between the leaves just jump to the bottom of the CM without interacting with the leaves and therefore cannot be recorded by the ray-tracing method. Although the geometry looks correct the later dosimetric simulations revealed the problem in the interleaf leakage. Hence the dosimetric comparisons are absolutely necessary for a complete validation of the CM.

The importance of modelling the details of the MLC lies in the fact that the effects and impact of the scattered photons and electrons, the transmission and leakage between the leaves as well as the field edge definition can be fully taken into account on dose calculations. The following dosimetric simulations allowed us to test these effects, comparing the MC data obtained using the CM with the measurements..

4.2.2.1 Output factors

A measurement of the dose in the phantom as a function of the field size is a necessary step in the validation process. The dose at a fixed point in the phantom depends on the size of the beam at the point of measurement increasing monotonically with the field size. The output factors depend on the particle scattering in the collimator (in our case the MLC), which varies with respect to the setting of the collimator. The agreement found between the measured and MC calculated OFs was within 2%, with the MC calculated values slightly higher than the measured values in the steep drop-off region, except for the 0.5 x 0.5 cm^2 field. For the smallest field the discrepancy is larger than in the rest of the OF profile.

4.2.2.2 The field size problem

The problem of the relation of the field size at isocenter and the mechanical position of leaves with "rounded" edges was introduced in Section 3.2.3. The comparison of the beam penumbra and different field edge precision measurements using MC simulations and dosimetric measurements were performed to answer this question.

In Figure 3.5 the agreement between the dosimetric measurements and the simulations is very good, especially in the penumbra section of the dose profile. Small discrepancies are observed in the shoulder region of certain profiles, where the dose scored by MC simulations is higher than the dosimetric measurement. Nevertheless the agreement in these cases is within 3 %. A mean penumbra of 3.2 mm for all X profiles was obtained from the films, and in the case of the MC simulations this value corresponds to 3.0 mm.

In Figure 3.6 it can be seen that the difference between the nominal field and the field obtained from the dosimetric films is greater than the difference obtained from the MC simulations in both cases. This can be attributed to the high uncertainty associated with the film handling during the measurements and processing. Nevertheless, the shape of the profile is very similar and in both cases we can see the same behaviour. In the case of the open position of the leaves (a), the difference increases with the field size, reaches a maximum around 4 cm from the beam central axis and then decreases. For the over travel case (b) we can see almost the same behaviour. However, at 4 cm a steep increase is observed.

In Figure 3.7 it can be seen that the profiles are similar between X_1 and X_2. Therefore, the difference in the accuracy found between the leaf banks is not significant. In the dosimetric characterization of the ModuLeaf (Hartmann and Föhlisch 2002) a mean deviation of 0.3 mm

was obtained between the nominal and the radiation field edge that is comparable to the mean value of 0.2 mm obtained from the MC simulations.

4.2.2.3 Interleaf leakage and transmission

The interleaf leakage profile obtained for the ModuLeaf is shown in Figure 3.8. From MC simulations a mean transmission value of 1.4% was obtained, with maximum values amount up to 1.9%. These values are close to the results obtained from dosimetric measurements, where a mean transmission of 1.3% and maximum values up to 1.8% were measured. In both cases the number of peaks due to interleaf leakage was found to be half the number of leaves. This effect can be attributed to the particular design of the leaf drive configuration in combination with the tilted leaf arrangement (Hartmann and Föhlisch 2002). Due to the tilted arrangement, the leaf displacement causes a larger attenuation thickness at every second leaf.

4.2.2.4 Uncertainties associated with the Monte Carlo simulations

The uncertainties associated with the MC simulations were in all cases smaller than the uncertainties obtained from the measurements. A standard deviation of 1% was obtained from the measurements of the OFs, for the MC simulations an uncertainty of 0.5% was calculated. For the measurements with films an uncertainty of less of 0.5 mm is estimated to be involved with this method and for MC simulations a value of 1% was calculated. For example, for the mean penumbra of the dose profiles this uncertainty corresponds to 0.03 mm. The small values obtained from the MC simulations allows a better analysis of the properties of the MLC studied, especially when the uncertainty of the techniques involved in the dosimetric measurements becomes comparable with the properties being studied. In Figure 3.7 the difference between the nominal leaf edge position and that obtained from the MC simulations is up to 0.035 mm approximately. This fact makes the measurement with films unsuitable for the evaluation of the actual position of the leaves edge and shows the advantage of the MC method, since it removes most of the uncertainties associated with the handling of measuring devices, such as dosimetric films. In Figure 3.8, it can also be seen that the MC shows small-size patterns of the interleaf dose leakage that cannot be seen with the film and suggests the use of a smaller-size dosimeter for this sort of configuration.

5 CONCLUSION

This dissertation presents the MC modeling of a ModuLeaf mini MLC using a modified EGSnrc MC platform. The DKFZ platform facilitates the entrance to the field of MC calculations in Medical Physics by means of its simplified structure and the use of the IVFC and its valuable debugging tool. These concepts were proved during the development of the MMLC CM for the ModuLeaf.

The MMLC CM is able to exactly model the details of the leaves, such as the leaf front edge shape and the tilting of the leaf bank. Therefore the next step was to investigate how well the dosimetric properties of the ModuLeaf can be reproduced using MC simulations. Dosimetric data as output factors, dose profiles, accuracy of field edge positioning, penumbra measurements, MLC interleaf leakage and transmission values have been compared with data obtained through simulations, using our MC platform. The uncertainty associated to the MC simulations is 1.0% and the analysis of the dose profiles showed an agreement between the dosimetric data and the MC simulations within 3%. A mean penumbra of 3.0 mm was obtained from MC that is comparable to the 3.2 mm obtained from film data. The difference between the nominal field size and the field size obtained from the dosimetric measurements showed a characteristic dependence on the position of the leaves, property that is also observed in the profiles obtained from the MC simulations. When studied specifically the deviation between the nominal and obtained position of the leaves, a mean deviation of 0.2 mm from MC simulations and 0.3 mm from the dosimetric characterization were obtained. A mean transmission value of 1.4% was calculated from MC that is close to the 1.3% obtained from dosimetric measurements.

The validation of the MMLC CM allowed us to prove and validate the development of the DKFZ platform. With this MC working tool we are able to do a more transparent study and development of the code. The debugging capabilities of IVFC simplified the detection of possible errors during the development of the MMLC CM and make easier the understanding of the simulation process. With our MC platform we are able to produce meaningful data that can be used for scientific purposes in radiotherapy studies. It was concluded that relations between leaf position and field edge depending on the shape of leaf ends can be investigated by this new CM with a higher accuracy as measurements. This demonstrates the usefulness of the new CM, which enhances the dosimetric characterization with respect to the measurements.

APPENDIX A: PARTICLE FLUENCE AS THE QUOTIENT OF THE TRACK-LENGTH DENSITY PER VOLUME

Consider the small irregular volume of interest ΔV in Figure A.1 which is exposed to a radiation field with particles of any arbitrary directional distribution. However, initially consider only those particles with direction $\vec{\Omega}$. If we consider "tubes" within the volume with length $h(x,y)$ and cross section da, depending on x and y, then the number of particles N with direction Ω is:

$$dN(\vec{\Omega}) = \Phi_{\vec{\Omega}}(\vec{\Omega}) d\vec{\Omega} da \tag{60}$$

Therefore the differential track length in direction $\vec{\Omega}$ is equal to:

$$dL_{\vec{\Omega}}(\vec{\Omega}) = dN(\vec{\Omega}) h(x,y) = \Phi_{\vec{\Omega}}(\vec{\Omega}) d\vec{\Omega} da \times h(x,y) \tag{61}$$

When Equation (60) is integrated over all directions the total differential track length is obtained:

$$dL = \int_{4\pi} \left[\Phi_{\vec{\Omega}}(\vec{\Omega}) d\vec{\Omega} da \times h(x,y)\right] d\vec{\Omega} \tag{62}$$

Equation (61) can be further modified knowing that $da \times h(x,y) = dV$

$$dL = \int_{4\pi} [\Phi_{\vec{\Omega}}(\vec{\Omega})\, d\vec{\Omega} dV] d\vec{\Omega} \tag{63}$$

From Equation (10), $\Phi = \int_{4\pi} [\Phi_{\vec{\Omega}}(\vec{\Omega})\, d\vec{\Omega}] d\vec{\Omega}$. Finally we obtain $dL = \Phi \cdot dV$ or $\Phi = dL/dV$.

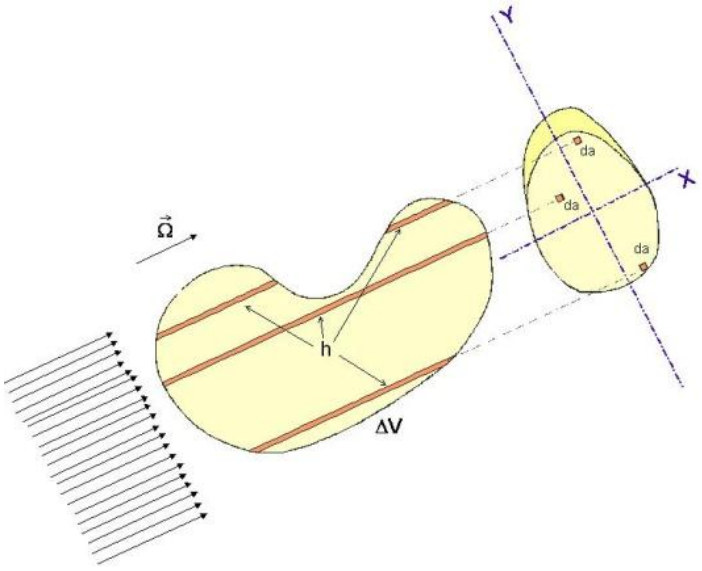

Figure A.1: Differential tracks of length dL and cross section da inside a small volume of interest ΔV that is exposed to a radiation field.

APPENDIX B: MAKEFILE TO CREATE A PARTICULAR BEAM CODE

```
MORTRAN_EXE = D:\\Monte_Carlo_Directory\\auxiliary\\mortran\\mortran3.exe
MORTRAN_DATA = D:\\Monte_Carlo_Directory\\auxiliary\\mortran\\mortran3.dat
MC_HOME = D:\\Monte_Carlo_Directory\\
accelerator = MMLC
ACCEL_HOME = $(MC_HOME)beam_build\\$(accelerator)\\
FORTRAN_FILE = beam_$(accelerator)
my_sources = $(MC_HOME)\\beam_build\\MORTRAN_beam_sources\\
SOURCES = \
      $(my_sources)egsnrc.macros \
      $(ACCEL_HOME)own_platform.macros \
      $(my_sources)timing.macros \
      $(my_sources)ranmar.macros \
      $(my_sources)transportp.macros \
      $(my_sources)beamnrc_user_macros.mortran \
      $(my_sources)phsp_macros.mortran \
      $(my_sources)sbsnrc_macros.mortran \
      $(ACCEL_HOME)$(accelerator)_macros.mortran \
      $(my_sources)egs_utilities.mortran \
      $(my_sources)beam_main.mortran\
      $(my_sources)beamnrc.mortran \
      $(my_sources)xvgrplot.mortran \
      $(ACCEL_HOME)$(accelerator)_cm.mortran \
      $(my_sources)get_inputs.mortran \
      $(my_sources)ranmar.mortran \
      $(my_sources)nrcaux.mortran \
      $(my_sources)machine.mortran \
      $(my_sources)egs_parallel.mortran \
      $(my_sources)egsnrc.mortran

$(FORTRAN_FILE).f: $(SOURCES) Makefile
      @echo Mortran compilation only (!!!) for $@
      @$(MORTRAN_EXE) -d $(MORTRAN_DATA) \
-f $(SOURCES) -o7 $@ -o8 $(FORTRAN_FILE).mortlst
```

APPENDIX C: MORTRAN CODE FOR THE MODULEAF MINI MLC

C.1 MMLC_cm.mortran

```
%E
/*****************************************************************************
 *
 *  $Id: MMLC_cm.mortran $
 *
 *****************************************************************************/
;
"*********************************************************************
"*********************************************************************
"
"                              ****************              ""toc:
"                              *               *              ""toc:
"                              *   MMLC     *              ""toc:
"                              *               *              ""toc:
"                              ****************              ""toc:
"
"
"*********************************************************************
"*********************************************************************
"I>
"I> Geometry of MMLC:
"I> **********************
"I>                                 top view
"I>
"I>            ---------------------------------------------
"I>                                |
"I>            ---------------------------------------------
"I>                          |           |
"I>           -----------------          ----------------------
"I>                      |        opening     |
"I>           ---------------     (IR=1)      -----------------
"I>                        |                  |
"I>           ------leaves--------      ------------------------
"I>                 (IR=2)      |     |
"I>           ---------------------------------------------------
"I>                               |
"I>           ---------------------------------------------------
"I>
;
"I>       cross-section through leaves taken perpendicular to leaves
"I>
"I>                                * ZFOCUS(1)
"I>                             .   .   .
"I>                         .    . .|. .    .
"I> - - - - - - -     .     .   . | .   .     .
"I> airgap(IR=3)  .      .     .  |  .     .      .
"I>     -------- ------ ------ ---|--- ------ ------ ----------
"I>   leaf     / leaf / leaf /leaf|leaf\ leaf  \ leaf \  leaf
"I>   1    /    2  /     3  /  4  | 5   \  6     \  7    \  8
"I>    /        /         /      |      \          \        \
"I>    -------- ---------- -------|------- ---------- --------- -
"I>                            Z-axis
;
"I>            cross-section through leaf taken || to leaf
"I>
"I>
"I>
"I>  - - - - - - - - - - - -    . | .
"I>  airgap(IR=3)              .  |  .
"I>     --------------------\     |     /----------------------
"I>                         |     |    |
"I>        Leaf B  (-ve)    |     |    |    Leaf A (+ve)
"I>                         |     |    |
"I>    --------------------/      |     \----------------------
"I>                            Z-axis
;
"I>   IR is the region number within the CM.  There are three local regions
"I>   shown above.
"I>
"I>
"I>            ------------------------------------------------------
"I>            |             Region            | Description         |
```

```
"I>            |-------------------------------|---------------------|
"I>            |  absolute       | local       |                     |
"I>            |-----------------|-------------|                     |
"I>            |                 |             |                     |
"I>            |IRSTART_MLC      | IR_MLC      |  as shown in above  |
"I>            |  +IR_MLC-1      |             |    (1 to 3)         |
"I>            |                 |             |                     |
"I>            |                 |             |                     |
"I>            ---------------------------------------------------------
"I>
;

" Subroutines:
" ************
"                      INPUT_$MMLC
"                      ISUMRY_$MMLC
"                      HOWFAR_$MMLC
"                      WHERE_AM_I_$MMLC
"
"       Called from BEAM's subroutines:
"                      INPUT
"                      ISUMRY
"                      HOWFAR
"
"       Subroutines called:
"                      WHERE_AM_I (a BEAM subroutine)
"
"*************************************************************************
"
"                             RESTRICTIONS ON USE/KNOWN BUGS
"                             ******************
"
"*************************************************************************
;
"
"                               INPUT FROM UNIT 5
"                               *****************
"
"I>
"I>  CARDS CM_$MMLC
"I>  ********************
"I>  -1  Dummy line to indicate start of CM
"I>
"I>   0  RMAX_CM(ICM_$MMLC) (F10.5):  Half-width of CM boundary (cm).
"I>
"I>   1  TITLE_$MMLC (60A1):  Title of CM.
"I>
"I>   2  ORIENT_$MMLC, MODE_$MMLC (3I5)
"I>
"I>            ORIENT_$MMLC = 0 for leaves parallel to Y direction
"I>                           = 1 for leaves parallel to X direction
"I>            MODE_$MMLC = 0 for single setting of leaf openings (static
"I>                           field)
"I>                         = 1 for dynamic mlc delivery--simulated leaf
"I>                           movement while beam is on
"I>                         = 2 for step-and-shoot delivery--beam off while
"I>                           leaf positions change
"I>
"I>   3  ZMIN_$MMLC (F15.0): Z of top of MLC (excluding airgap)
"I>
"I>   4  ZTHICK_$MMLC (F15.0): Thickness of the leaves ( z-axis (cm))
"I>
"I>   5  LEAFWIDTH_$MMLC, (15F15.0)
"I>
"I>   6  TOT_LEAF_$MMLC (I5): Number of leaves
"I>
"I>   7  LEAFGAP_$MMLC (F15.5) : The width of the interleaf air gap
"I>            at ZMIN_$MMLC.
"I>
"I>   8  ZFOCUS_$MMLC (F15.5): Focal point on Z-axis of leaf sides
"I>                        imaginary lines drawn extending the slopes of
"I>                        the leaf sides will all intersect the Z-axis
"I>                        at this point)
"I>
"I>       Note restriction: ZFOCUS_$MMLC < ZMIN_$MMLC or
"I>                                > ZMIN_$MMLC + ZTHICK_$MMLC
"I>
"I>   9  HEDGEUP_O_$MMLC, HEDGEDOWN_O_$MMLC, WEDGEUP_O_$MMLC,
```

```
"I>            WEDGEDOWN_O_$MMLC, HEDGEUP_U_$MMLC, HEDGEDOWN_U_$MMLC,
"I>            WEDGEUP_U_$MMLC, WEDGEDOWN_U_$MMLC (15F15.0):
"I>            Leaves dimensions in the direction along leaf movement.
"I>
"I>       10  YGROOVEUP_U_$MMLC, WGROOVEDOWN_U_$MMLC, WGROOVEUP_U_$MMLC,
"I>            YGROOVEDOWN_O_$MMLC, WGROOVEUP_O_$MMLC, WGROOVEDOWN_O_$MMLC (15F15.0):
"I>            Leaves dimensions in the direction perpendicular to leaf movement.
"I>
"I>       11 ZGROOVEDOWN_U_$MMLC, HGROOVEUP_U_$MMLC, HGROOVEDOWN_U_$MMLC,
"I>            ZGROOVEUP_O_$MMLC, HGROOVEUP_O_$MMLC, HGROOVEDOWN_O_$MMLC (15F15.0):
"I>            Leaves dimensions along Z axis.
"I>
"I>       12  ZTILT_$MMLC, TILT_$MMLC (15F15.0): Z and angle of leaves tilting.
"I>
"I>      If MODE_$MMLC=0 (static field):
"I>
"I>     Repeat 14a until opening coordinates of all leaves are defined once.
"I>     Leaves are numbered 1,2,...TOT_LEAF_$MMLC, where numbering goes from
"I>     leaf 1 to leaf TOT_LEAF_$MMLC. Convention is lower to upper or
"I>     left to right depending on ORIENT_$MMLC i.e from negative to
"I>     positive.  Note that for dynamic or step-and-shoot simulations, these
"I>     are the default coordinates, used unless specified otherwise in the
"I>     file of leaf opening data input in line 14a (see below).
"I>
"I>  13a  NEG_$MMLC, POS_$MMLC, NUM_$MMLC (2F15.5,I5)
"I>
"I>     NEG_$MMLC:  Min. Y (ORIENT_$MMLC=0) or X (ORIENT_$MMLC=1)
"I>                    of front opening in leaf I (ie the opening at
"I>                    ZMIN_$MMLC) if ENDTYPE=1, or of rounded end
"I>                    of leaf I if ENDTYPE=0.
"I>     POS_$MMLC:  Max. Y (ORIENT_$MMLC=0) or X (ORIENT_$MMLC=1)
"I>                    of front opening in leaf I if ENDTYPE=1, or of
"I>                    rounded end of leaf I if ENDTYPE=0.
"I>     NUM_$MMLC:  Apply NEG_$MMLC and POS_$MMLC to leaves
"I>                    I,...,I+NUM_$MMLC-1.  Defaults to 1 if set <=0.
"I>                    Defaults to TOT_LEAF_$MMLC-I+1 if set >
"I>                    TOT_LEAF_$MMLC-I+1.
"I>
"I>  If MODE_$MMLC=1 or 2 (dynamic delivery or step-and-shoot delivery):
"I>
"I>  13b  mlc_file (A80)
"I>
"I>         mlc_file:  The full name of the file containing leaf opening
"I>                    data.  The format of the file contents is as follows:
"I>
"I>      MLC_TITLE (A80)
"I>      NFIELDS_$MMLC (I10)
"I>      FOR I=1,NFIELDS_$MMLC[
"I>       INDEX_$MMLC(I) (F15.0)
"I>       NEG_$MMLC, POS_$MMLC, NUM_$MMLC (2F15.0,I5) -- repeat this
"I>                                                                line until
"I>                                                                coordinates
"I>                                                               for all leaves
"I>                                                                have been
"I>                                                                defined for
"I>                                                                field I.
"I>      ]
"I>
"I>      where:
"I>
"I>             MLC_TITLE:  A title line
"I>      NFIELDS_$MMLC:  Total number of fields
"I>     INDEX_$MMLC(I):  Index of field I. 0 <= INDEX_$MMLC(I) <= 1 and
"I>                         INDEX_$MMLC(I) > INDEX_$MMLC(I-1).  This
"I>                         number is compared to a random number on (0,1) at
"I>                         the start of each history; if the random number is
"I>                         <= INDEX_$MMLC(I), then field I is used.
"I>          NEG_$MMLC:  Min. Y (ORIENT_$MMLC=0) or X (ORIENT_$MMLC=1)
"I>                         of front opening in leaf (ie the opening at
"I>                         ZMIN_$MMLC) if ENDTYPE=1, or of rounded end
"I>                         of leaf if ENDTYPE=0 for leaf J in field I.
"I>          POS_$MMLC:  Max. Y (ORIENT_$MMLC=0) or X (ORIENT_$MMLC=1)
"I>                         of front opening in leaf if ENDTYPE=1, or of
"I>                         rounded end of leaf if ENDTYPE=0 for leaf J in
"I>                         field I.
"I>          NUM_$MMLC:  Apply NEG_$MMLC and POS_$MMLC to leaves
"I>                         J,...,J+NUM_$MMLC-1.  Defaults to 1 if set <=0.
"I>                         Defaults to TOT_LEAF_$MMLC-J+1 if set >
```

```
"I>                          TOT_LEAF_$MMLC-J+1.
"I>
"I>    Note that the inputs NEG_$MMLC, POS_$MMLC and NUM_$MMLC have
"I>    the same meanings as in 14a (static field inputs) but that they must
"I>    now be repeated for every field I.
"I>
"I>  14   ECUT, PCUT, DOSE_ZONE, IREGION_TO_BIT in opening(s) and
"I>                                              air gaps (2F15.5,2I5)
"I>
"I>        ECUT, PCUT:  Cutoff energies for electrons and photons.
"I>         DOSE_ZONE:   Dose scoring flag, 0 to not score dose
"I>    IREGION_TO_BIT:   Bit number associated with this region
"I>
"I>  15  MED_IN (24A1):  Medium in opening(s) and air gaps
"I>                       used to set MED_INDEX.
"I>
"I>  16  ECUT, PCUT, DOSE_ZONE, IREGION_TO_BIT in leaves, IGNOREGAPS_$MMLC
"I>                                                      (2F15.0,3I5):
"I>
"I>        ECUT, PCUT:  Cutoff energies for electrons and photons.
"I>         DOSE_ZONE:   Dose scoring flag, 0 to note score dose
"I>    IREGION_TO_BIT:   Bit number associated with this region
"I>        IGNOREGAPS:   If set to 1, ignore all air gaps and driving screw
"I>                      holes when doing range
"I>                      rejection in leaf material when the particle X position
"I>                      is < min X of all leaf openings (not including leaf
"I>                      ends) or > max X of leaf openings (not including ends)
"I>                      (ORIENT_$MMLC=1) or if the particle Y position
"I>                      is < min Y of all leaf openings (not including leaf
"I>                      ends) or > max Y of leaf openings (not including ends)
"I>                      (ORIENT_$MMLC=0). This approximation is designed
"I>                      to make range rejection more efficient deep in the
"I>                      leaves, while still preserving accurate transport
"I>                      in the leaf ends.  Note that if you have significant
"I>                      air gaps between leaves or are concerned with the
"I>                      effects of the driving screw holes it is recommended
"I>                      that you not use this option (ie run with the default
"I>                      setting of 0).
"I>
"I>  17  MED_IN (24A1):  Medium of leaves,
"I>                       used to set MED_INDEX.
"I>
"I>  18  ECUT, PCUT, DOSE_ZONE, IREGION_TO_BIT in driving screw holes
"I>                                                     (2F15.5,2I5):
"I>
"I>        ECUT, PCUT:  Cutoff energies for electrons and photons.
"I>         DOSE_ZONE:   Dose scoring flag, 0 to note score dose
"I>    IREGION_TO_BIT:   Bit number associated with this region
"I>
"I>  19  MED_IN (24A1):  Medium in driving screw holes,
"I>                       used to set MED_INDEX.
"I>
"I>   Example
"I>   *******
"I>*********** start of CM MMLC with identifier MMLC ***********
"I>21.0, RMAX
"I>Siemens ModuLeaf mini MLC
"I>1, 0,  ORIENT, MODE
"I>60.86, ZMIN
"I>7.0, ZTHICK
"I>0.130
"I>40
"I>0.02353, LEAFGAP
"I>-1.4334, ZFOCUS of leaf sides
"I>2.525, 1.975, 0.13887, 0.1185, 1.975, 2.525, 0.10862, 0.1515
"I>0.035, 0.09, 0.07, 0.030, 0.09, 0.07
"I>0.3, 0.3, 0.2, 0.3, 0.2, 0.3
"I>64.63271, 0.76, ZTILT, TILT
"I>0.0, 0.0, 1
"I>-3.2354, 3.2354, 38
"I>0.0, 0.0, 1
"I>0.512, 0.01, 0, 0,
"I>AIR512ICRU
"I>0.512, 0.01, 0, 0,
"I>W512ICRU
;
"************************************************************************
"************************  ERROR CONDITIONS  ***************************
```

```
"***************************************************************************
"
"                               SIMULATION PARAMETERS
"                               *********************
"
" Geometry checks:
" ****************
"
"   1)  Overlapping component modules
"
"***************************************************************************
;
%E "Start of subroutine HOWFAR_$MMLC "
"***************************************************************************
"*******************Component Module MMLC ****************************
"***************************************************************************
"
"                           Subroutine HOWFAR_$MMLC
"                           ***********************
"
" HOWFAR routine for stacked planar media.
"
" Determine if current region number is within component module MMLC,
" evaluate DIST, distance to region boundary along current trajectory.
" USTEP must not exceed DIST.
"
"   There are N_$MMLC local regions + an air gap (if present):
"
"  local             absolute                                    description
"-------------- ---------------------------------------- ---------------
"IR_$MMLC IR_start_CM(ICM_$MMLC)+IR_$MMLC-1 exclude front air gap
"-------------- ---------------------------------------- ---------------
"   Geometrical co-ordinates, as set in INPUT_$MMLC are:
"
"   ZFRONT_$MMLC     front of CM (upstream surface, air region)
"   ZMIN_$MMLC       front of cone
"   ZBACK_$MMLC      back of cone
"   RMAX_$MMLC       radius of last cylinder (largest)
"
"***************************************************************************
;
;SUBROUTINE HOWFAR_$MMLC;

"V>$GEO_SHIFT_1_(#)
"V>=============
"V>{p1} the value to compare with ustep
"V> if {p1}+1.0e-5 < ustep shift it, otherwise no shift
;
REPLACE {$GEO_SHIFT_1_(#)} WITH {
  ;IF({P1}~=0.00 ) [
    IF({P1}+1.0E-5< USTEP) [{P1}={P1}+1.0E-5;]
   ]
   ELSE [{P1}={P1}+1.0E-5;];

}

;IMPLICIT NONE;
;COMIN/CMs,CM_$MMLC,EPCONT,STACK,USER,EGS-IO/;
"T>
"T>*********************************
"T>TYPE DECLARATIONS FOR HOWFAR_$MMLC
"T>*********************************
"T>
INTEGER
   COUNT,
   IRL,         "T>local region number (absolute), required by HOWNEAR macro
   I, J,         "T>loop control
   REGION_$MMLC,     "T>region number within CM (relative)
   NEWREGION_$MMLC,     "T>region number within CM (relative)
   NX,NY,NZ,         "T> Subindices for region
   OUTOFCMFLAG,     "T> Flag to denote particle out of CM
   OUTOFMLCFLAG,    "T> Flag to denote particle out of MLC
   LEAFIS,          "T> Leaf number
   LHS,RHS,     "T> Macro variables to find + and - closest boundary index
   ZLHS,ZRHS,
   IDIR;         "T> direction sign, used in call to where_am_i

DOUBLE PRECISION
```

```
   DIST,      "T>Distance to z boundary along current particle trajectory
   UVL(2),    "T>temporary variable
   TRYY1,TRYY2,
   XoN,XoP,    "T> Negative and Positive leaf centers (rounded leaf)
   Zo,         "T> z position of rounded leaf tip
   XP,XN,YP,YN,ZP,ZN,    "T>+ and - distances in x,y, z directions
   XDIST,YDIST,ZDIST,    "T> x,y, Z distances to nearest boundaries
   DIST1,DIST2,DIST3,    "T> Variables to control particle propagation
   STEP2,                "T>  "
   TLHS,TRHS,            "T> temporary macro variables like lhs,rhs
   DISCRIMINANT,         "T> Variable for rounded leaf
   TEMP,TEMP1, TEMP2, TEMP3, TEMP4, TEMP5, HOLE,
   XN1, XN2, XN3, XP1, XP2, XP3, "T> temporal distance variables"
   STEP_UNIT,
   XYL(2),     "T> rearranged x and y coordinates
   XYFL(2),
   ZFL;

$REAL one;
parameter (one = 1);

" prepare the local variables "
"============================="

IRL = IR(NP); "local region number (absolute)"
IR_$MMLC = IRL - IRSTART_$MMLC + 1;   "rel. local region number"
IF(ORIENT_$MMLC=1) [
    XYL(1)=Y(NP); XYL(2)=X(NP);UVL(1)=V(NP);UVL(2)=U(NP);]
ELSE[ XYL(1)=X(NP); XYL(2)=Y(NP);UVL(1)=U(NP);UVL(2)=V(NP);]
;

STEP_UNIT=0.0;
COUNT = 0;
OUTOFCMFLAG=0;
OUTOFMLCFLAG=0;

" Boundary-crossing check
" **********************
"
" Determine if current region number is within component module ,if so
" evaluate DIST, distance to region boundary along current trajectory.
" USTEP must not exceed DIST.
"

" the following block double check:
"================================
"      1. the particle is out of the CM or regions in z direction
"      2.                          the regions in xy directions
"     if so reset the ir #, and print out the warning message.
"comment: this block is not time consumming, can be kept in the final version
"=======
"
"  now do the air gap check if existed.
"====================================
"

IF(N_GAP_$MMLC=1 & IR_$MMLC=3 ) [
   IF(W(NP)>0.0) ["Particle going forward"
      DIST = (ZMIN_$MMLC - Z(NP))/W(NP); "distance to front of CM"
      IF(DIST <= 0.0) [
          USTEP=0.0;
          $MMLC_FIND(IR_$MMLC, 0.0);
          IF( IR_$MMLC=1 ) "in air"
            [IRNEW =IRSTART_$MMLC; RETURN;]
          ELSE[IRNEW =IRSTART_$MMLC+1; RETURN;]
       ] "double check if a particle is out of the AIR GAP"
    ]
   ELSEIF(W(NP)<0.0) [ "particle going backward"
      DIST = (ZFRONT_$MMLC - Z(NP))/W(NP); "distance to front of CM"
      IF(DIST <= 0.0) [
          USTEP=1.E-16;
          CALL WHERE_AM_I(ICM_$MMLC,-1);
          RETURN;
      ] " double check if a particle is out of the CM"
    ];
 ]
ELSEIF(IR_$MMLC=1 | IR_$MMLC=2) [
   IF(W(NP) > 0.0) ["particle going forward"
```

```
      DIST = (ZMAX_$MMLC - Z(NP))/W(NP); "distance to back of CM"
      IF(DIST>USTEP) [ STEP_UNIT=USTEP; ]
      ELSE [ STEP_UNIT=DIST;  ]
      IF(DIST <=0.0) [
          USTEP=1.E-16;"ensures call to AUSGAB on leaving CM"
          CALL WHERE_AM_I(ICM_$MMLC,1);
          RETURN;
       ] "double check if a particle is out of the CM"
    ]
   ELSEIF(W(NP) < 0.0) [
     "particle going backward"
      DIST = (ZMIN_$MMLC - Z(NP))/W(NP); "distance to back of CM "
      IF(DIST>USTEP) [ STEP_UNIT=USTEP;  ]
      ELSE [ STEP_UNIT=DIST; ]
      IF(DIST <= 0.0) [
          USTEP=0.0;
          IF(N_GAP_$MMLC = 1)
            [IRNEW =IREND_$MMLC; RETURN; ]
          ELSE[ USTEP=1.E-16; CALL WHERE_AM_I(ICM_$MMLC,-1); RETURN; ]
      ] " double check if a particle is out of the main body to air gap"
    ]
   ELSE[STEP_UNIT=USTEP;]  " for w(np)=0.0 case"
 ];

" end of z direction check
"=========================

$MMLC_FIND(REGION_$MMLC,0.0);

IF(REGION_$MMLC=3) ["Particle is in the air gap"
  REGION_$MMLC=IRSTART_$MMLC-1+REGION_$MMLC;
   IF(W(NP) > 0.0)[
    IF(DIST <= USTEP ) ["particle to be moved to region boundary"
      $GEO_SHIFT_1_(DIST);
      USTEP = DIST;
      $MMLC_FIND(NEWREGION_$MMLC,USTEP);
      IF( NEWREGION_$MMLC=1) [IRNEW =IRSTART_$MMLC; RETURN;]
      ELSE[IRNEW =IRSTART_$MMLC+1; RETURN;]
    ]
    ELSE[ RETURN;];
    ] "end of particle going forward"
    ELSEIF(W(NP) < 0.0) ["particle going backward"
      IF(DIST <= USTEP ) ["particle to be moved to region boundary"
        $GEO_SHIFT_1_(DIST);
        USTEP = DIST;
        CALL WHERE_AM_I(ICM_$MMLC,-1);
        RETURN;
        ]
      ELSE [RETURN;]
    ] " end of going backward"
    ELSE[RETURN;];   " W=0.0 CASE "
] ;" end of region 3"

"New Y parametrization"
IF(OUTOFMLCFLAG=1) [
  STEP2=STEP_UNIT;
  TEMP1=MYREG_$MMLC(1,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(1,1)-XYFL(1);
  IF((UVL(1)-MYREG_$MMLC(1,1)*W(NP))~=0.0 ) [
     TEMP1=TEMP1/(UVL(1)-MYREG_$MMLC(1,1)*W(NP));
   ]
  ELSE [ TEMP1 = -1000.00 ];

  TEMP2=MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(Z(NP)-ZTILT_$MMLC)+
        NYREG_$MMLC(TOT_LEAF_$MMLC,6)-XYFL(1);
  IF( (UVL(1)-MYREG_$MMLC(TOT_LEAF_$MMLC,6)*W(NP))~=0.0 ) [
              TEMP2=TEMP2/(UVL(1)-MYREG_$MMLC(TOT_LEAF_$MMLC,6)*W(NP));
  ]
  ELSE [ TEMP2 = -1000.00 ];

  IF( STEP2>=0.0)[
    TEMP=STEP2;
    IF( TEMP1>=0 ) [TEMP=MIN(TEMP,TEMP1);];
    IF( TEMP2>=0 ) [TEMP=MIN(TEMP,TEMP2);];
  ]
  ELSEIF ( (TEMP1>0.0) & (TEMP2>0.0) ) [TEMP=MIN(TEMP1,TEMP2);]
  ELSE [TEMP=MAX(TEMP2,TEMP1);];

  STEP2 = TEMP;
```

```
  $GEO_SHIFT_1_(STEP2);
  $MMLC_FIND(IRNEW,STEP2);
  IRNEW=IRNEW+IRSTART_$MMLC-1;
  USTEP=STEP2;
  RETURN;
 ]
;
$MMLC_MINDISTANCE(DIST1);

IF((DIST1>=STEP_UNIT) | (DIST1<0)) [
    STEP2=STEP_UNIT;
    $GEO_SHIFT_1_(STEP2);
    $MMLC_FIND(IRNEW,STEP2);
    USTEP=STEP2;
    IF(OUTOFCMFLAG=1)[ "call where_am_i"
       IDIR=SIGN(one,W(NP));
       CALL WHERE_AM_I(ICM_$MMLC,IDIR);
       RETURN;
    ]
    ELSE[
      IRNEW=IRNEW+IRSTART_$MMLC-1;
      RETURN;
    ]
]
ELSEIF( (STEP_UNIT>DIST1) & (DIST1>=0)) [
    REGION_$MMLC=REGION_$MMLC+IRSTART_$MMLC-1;
    LOOP[
      DIST1=DIST1+1.0E-5; "we have to shift it regardless to avoid "
                          "infinite loops"
      IF(DIST1>=STEP_UNIT)[EXIT;]
      ELSE[
        $MMLC_FIND(NEWREGION_$MMLC,DIST1);
        NEWREGION_$MMLC=NEWREGION_$MMLC+IRSTART_$MMLC-1;
        IF( OUTOFCMFLAG=1 ) [ EXIT;]
        ELSEIF( (NEWREGION_$MMLC~=REGION_$MMLC) | OUTOFMLCFLAG=1 )[
        USTEP = MIN(USTEP,DIST1); "take min. because we shifted DIST1"
        "start ray-tracing for debugging geometry"
        IF(((NEWREGION_$MMLC-IRSTART_$MMLC+1)=2 &
               (REGION_$MMLC-IRSTART_$MMLC+1)=1)|
           ((NEWREGION_$MMLC-IRSTART_$MMLC+1)=1 &
               (REGION_$MMLC-IRSTART_$MMLC+1)=2)) [
           WRITE(82,*) Y(NP)+USTEP*V(NP),Z(NP)+USTEP*W(NP);
           WRITE(80,*) X(NP)+USTEP*U(NP),Y(NP)+USTEP*V(NP);
           WRITE(81,*) X(NP)+USTEP*U(NP),Z(NP)+USTEP*W(NP);];
        IRNEW = NEWREGION_$MMLC;
        RETURN;
       ]
       ELSE [
         $MMLC_MINDISTANCE(DIST2);
         DIST1=DIST1+DIST2;
       ];
      ]
    ];
;
"if it gets here, then it exited either because OUTOFCMFLAG=1"
"or because DIST1 >= STEP_UNIT, in either case DIST1 >= STEP_UNIT"
;
    DIST1=STEP_UNIT;
    $GEO_SHIFT_1_(DIST1);
    USTEP = DIST1;

    IF( OUTOFCMFLAG=0 ) [
      $MMLC_FIND(NEWREGION_$MMLC,DIST1);
      "still have to check if we have left the CM in case STEP_UNIT"
      "was the distance to the bottom of the CM"
      IF(OUTOFCMFLAG=1)[
        IDIR=SIGN(one,W(NP));
        CALL WHERE_AM_I(ICM_$MMLC,IDIR);
        RETURN;
      ]
      ELSE["not leaving CM"
        NEWREGION_$MMLC=NEWREGION_$MMLC+IRSTART_$MMLC-1;
        IRNEW = NEWREGION_$MMLC;
        RETURN;
      ]
    ]
    ELSE[ "call where_am_i"
```

```
       "since flag is based on distance to Z bdy, DIST1 is >= STEP_UNIT now"
       IDIR=SIGN(one,W(NP));
       CALL WHERE_AM_I(ICM_$MMLC,IDIR);
       RETURN;
    ]

]; "End of IF loop for step_unit>dist1

;
"   end of HOWFAR_$MMLC
"   ===================
"
RETURN;
END; "End of subroutine HOWFAR_$MMLC"
%E "Start of WHERE_AM_I_$MMLC "
"*************************************************************************
"
"                          Subroutine WHERE_AM_I_$MMLC
"                          ***************************
"
" WHERE_AM_I routine for a stacked right cylinder slabs.
"
" WHERE_AM_I_$MMLC determines the new region number when a particle
" traverses a component module boundary.  The scheme is as follows:
"
"      Whenever a particle is to be transported to a component module
"      boundary in HOWFAR, the subroutine WHERE_AM_I is called.  The
"      current component module and particle direction (backwards or
"      forwards) are transferred to WHERE_AM_I in the CALL statement.
"      WHERE_AM_I determines which component module the particle is
"      about to enter and calls the WHERE_AM_I_$MMLC subroutine for
"      that component module, transferring the particle direction.
"      The region number that the particle is about to enter is
"      determined in WHERE_AM_I_$MMLC from the knowledge of which
"      surface the particle is entering through (front if IDIR=1,
"      back if IDIR=-1) and the (X,Y) coordinates of the particle.
"      The current particle being transported is NP (in /STACK/).
"
"*************************************************************************

;SUBROUTINE WHERE_AM_I_$MMLC(IDIR);
;
;IMPLICIT NONE;
;COMIN/CM_$MMLC,EPCONT,STACK,CMs,USER/;
"T>
"T>************************************
"T>TYPE DECLARATIONS FOR WHERE_AM_I_$MMLC
"T>************************************
"T>
INTEGER NX,NY,NZ,   "T>Indices of subregions
        I,
        OUTOFCMFLAG, "Flags to denote out of CM and
        OUTOFMLCFLAG, " MLC respectively
        LEAFIS,       " Leaf number
        IDIR;  "T>direction of particle, +1=forward, -1=backward
DOUBLE PRECISION   XYL(2), XYFL(2), ZFL, UVL(2), XoN, XoP, Zo, TEMP1, TEMP2,
                   TEMP3, TEMP4, HOLE;

IF(ORIENT_$MMLC=1)
    [ XYL(1)=Y(NP); XYL(2)=X(NP);UVL(1)=V(NP);UVL(2)=U(NP);]
ELSE[ XYL(1)=X(NP); XYL(2)=Y(NP);UVL(1)=U(NP);UVL(2)=V(NP);];
IF (IDIR=1) [
 "particle entering this CM through front face (upstream)"
   IF(N_GAP_$MMLC = 0) [ "no air gap this CM"
       IF(IRSTART_$MMLC=2)   " the first CM"
         [$MMLC_FIND(IR_$MMLC,0.0); ]
       ELSE[$MMLC_FIND(IR_$MMLC, USTEP);]
       IF(IR_$MMLC=1)
         [IRNEW=IRSTART_$MMLC; RETURN;]
       ELSE[ IRNEW=IRSTART_$MMLC+1; RETURN; ]
    ]         " end of the no air gap case"
   ELSE [ "this CM has an air gap at the front"
      IRNEW = IREND_$MMLC;
        ]
 ]

ELSE [ "particle entering CM through back face (downstream)"
   $MMLC_FIND(IR_$MMLC, USTEP);
```

```
   IF(IR_$MMLC=1)
     [IRNEW=IRSTART_$MMLC; RETURN;]
   ELSE[ IRNEW=IRSTART_$MMLC+1; RETURN; ]
 ];

RETURN;
;
END; "End of subroutine WHERE_AM_I_$MMLC"
%E "Start of subroutine INPUT_$MMLC "

"***********************************************************************

"                          Subroutine INPUT_$MMLC
"

"***********************************************************************
"***********************************************************************
"  A CM input subroutine for a series of 2 or more slabs.
"
"  It must fill all parameters in COMMON/CMs/ associated with this CM.
"
"  Routine prints error messages on unit 6 for
"      format error on input
"      end of file hit
"      error in logic of input file
"
"  The format of the input is presented in the section INPUT FROM UNIT 5
"   in the above documentation.
"
"***********************************************************************

;SUBROUTINE INPUT_$MMLC;
;

;IMPLICIT NONE;
;COMIN/ BOUNDS,CMs,CM_$MMLC,GEOM,IO_INFO,MEDIA,MISC,SCORE,USER,EGS-IO/;

"  ************************************************************ "
"              TYPE DECLARATIONS FOR INPUT_$MMLC
"  ************************************************************ "

DOUBLE PRECISION  NEG_$MMLC,     "T>Leaf B
                  POS_$MMLC,     "T>Leaf A
                  TEMP1,
                  TEMP2,
                  TEMP3,
                  TEMP4;

INTEGER I,J,K,L,        "T>DO loop indices
        IRA,            "T>Absolute region number
        MED_FLAG,       "T>flag used by media-sort macro $MED_INPUT
        MED_INDEX,   "T>medium index, set after medium sort by $MED_INPUT
     NUM_$MMLC, "T>number of adjacent leaves with same opening coordinates
       MIN_INDEX,   "T>index of leaf with min. opening coordinate
       MAX_INDEX,   "T>index of leaf with max. opening coordinate
       mlc_unit,    "T>unit no. to assign mlc_file
       egs_get_unit; "T>egs function used to assign unit to 2mlc_file
CHARACTER*80 mlc_file, "T>name of file containing opening data for dynamic and
                      "  step-and-shoot sequences
             MLC_TITLE; "T> title line in mlc_file
$REAL INDEXTMP; "T>temporary input variable for field indices in dynamic and
                "  step-and-shoot simulations
" ************************************************************ "

"                 STEP I : INITIALIZE PARAMETERS
"                          ======================

" ************************************************************ "


"I. GET THE TITLE "
"================ "
;
EPS=0.00001;
OUTPUT;(/' Next component is a Moduleaf MLC'/' Title: ',$);
MINPUT ($MMLC) TITLE_$MMLC;(60A1);
          "MINPUT is a replacement macro with EOF and
          "ERR branching to :EOF_{P1}: and :ERR_{P1}:
```

```
OUTPUT TITLE_$MMLC;(' ',60A1);
            "OUTPUT is a replacement macro which writes to
            "unit 5.  Used here for echo of user input

"II. CHOOSE THE MMLC ORIENTATION "
"============================== "

OUTPUT; (' Input leaf orientation (0=parallel to y, 1=parallel to x) and '/
         ' mode of the MLC'/' :',$);
MINPUT ($MMLC) ORIENT_$MMLC, MODE_$MMLC; (3I5);
OUTPUT ORIENT_$MMLC, MODE_$MMLC; (2I5/);
IF(ORIENT_$MMLC~=1) [
      ORIENT_$MMLC=0;
      OUTPUT;(/' Orientation defaults to 0.'/);
 ];
IF(MODE_$MMLC<0 | MODE_$MMLC>2) [
      MODE_$MMLC=0;
      OUTPUT;(/' Mode of leaf opening input defaults to 0.'/);
 ];

"III. DESIGNATE REGION NUMBERS "
"============================== "

N_$MMLC = 2; "Number of regions in this CM (excluding front air gap)"
ICM_$MMLC = ICM;      "CM index for this component module
IRSTART_$MMLC = IR_start_CM(ICM_$MMLC);
                      "Index of first region in this CM,
                      "set by previous CM or in MAIN if ICM=1
IERR_GEOM(ICM_$MMLC) = 0; "Geometry-checking flag, 0 if no error detected

"IV. GET DISTANCE FROM THE REFERENCE PLANE, z=0 "
"============================================== "
;
OUTPUT; (' Z position of top of compact MLC (>=0) : ',$);
;MINPUT ($MMLC) ZMIN_$MMLC;(F12.5);
OUTPUT ZMIN_$MMLC;(F12.5/);

"ZMIN_$MMLC is shifted 0.5 cm towards the target. As the leaves are arranged"
"with respect to ZFOCUS, some of the ends are above the user defined ZMIN, and"
"therefore above the upper limit of the CM. Shifting this boundary solves"
"the problem."

ZMIN_$MMLC=ZMIN_$MMLC-0.5;

IF(Z_min_CM(ICM_$MMLC)>ZMIN_$MMLC) [
   IF(ICM_$MMLC=1) [
      Z_min_CM(ICM_$MMLC)=ZMIN_$MMLC;
      OUTPUT ICM_$MMLC, Z_min_CM(ICM_$MMLC);
       (//' ***WARNING IN CM ',I4,' ($MMLC):'/
          'Z_min_CM(1)  > distance to front of MLC '/
          'Z_min_CM(1) reset to ',F8.5,' cm'//);
      WRITE(IOUTLIST,
         '(//'' ***WARNING IN CM '',I4,'' ($MMLC):''/
         '' Z_min_CM(1)  > distance to front of collimator''/
         '' Z_min_CM(1) reset to '',F8.5,'' cm''//)')
         ICM_$MMLC, Z_min_CM(ICM_$MMLC);
     ]
    ELSE[
      OUTPUT ICM_$MMLC;
       (//' ***** WARNING WARNING WARNING *****'/
         ' ***ERROR IN CM ',I4,' ($MMLC):'/
         ' Overlaps with previous CM'/
         ' Error will be propagated'//);
      WRITE(IOUTLIST,
         '(//'' ***WARNING IN CM '',I4,'' ($MMLC):''/
         '' Overlaps with previous CM''//)')
         ICM_$MMLC;
      IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
     ]
 ]
;
" ********************************************************** "

"              STEP TWO : GET MMLC GEOMETRY INFORMATION
"                         ============================

" ********************************************************** "
```

```
" I. THICKNESS OF LEAVES "
" ======================= "
;

OUTPUT; (' MLC Leaf thickness (cm):',$);
MINPUT ($MMLC) ZTHICK_$MMLC; (F15.0);
OUTPUT ZTHICK_$MMLC;(F15.5, ' cm'/);

" Validate the user-input thickness "

IF(ZTHICK_$MMLC<0.0) [
    OUTPUT ICM_$MMLC;(//' ***ERROR IN CM ',I4,' ($MMLC):'/
                       ' ZTHICK < 0.0'//);
    IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
 ];

"ZMAX_$MMLC is shifted backwards the target in order to take into account the"
"shift of ZMIN and the shift between the leaves"
ZMAX_$MMLC = ZMIN_$MMLC + ZTHICK_$MMLC + 1.5;
;

" II. Input geometries of leaves"
" ============================================================== "
;
OUTPUT; (/'Input the leaves width (cm): ',$);
  "leaf width defined at ZMIN"
MINPUT ($MMLC) LEAFWIDTH_$MMLC; (F15.0);
OUTPUT LEAFWIDTH_$MMLC; (F12.5);

TOT_LEAF_$MMLC=0;
K=0;

OUTPUT;
(/' For MMLC leaves input number of leaves: '/);
MINPUT ($MMLC) TOT_LEAF_$MMLC; (2I5);
OUTPUT TOT_LEAF_$MMLC; (2I5);

IF(TOT_LEAF_$MMLC>$MAXLEAF) [
    TOT_LEAF_$MMLC=$MAXLEAF;
    OUTPUT ICM_$MMLC,TOT_LEAF_$MMLC;
            (//' ***ERROR IN CM ',I3,' ($MMLC)'/
             ' Total # of leaves > max allowed'/
             ' Total # of leaves reduced to ',I3,' for now'//);
    IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
];
IF(TOT_LEAF_$MMLC<=0) [
    TOT_LEAF_$MMLC=1;
    OUTPUT ICM_$MMLC,TOT_LEAF_$MMLC;
            (//' ***ERROR IN CM ',I3,' ($MMLC)'/
             ' Total # of leaves <= 0'/
             ' Total # of leaves set to ',I3,' for now'//);
    IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
 ];

"III. INTER-LEAF AIR GAP "
"======================= "

OUTPUT; (/'Input the inter-leaf air gap(>=0.0) : ',$);
  "leaf gap defined at ZMIN"
MINPUT ($MMLC) LEAFGAP_$MMLC; (F15.0);
OUTPUT LEAFGAP_$MMLC; (F12.5);

IF (LEAFGAP_$MMLC<0.0) [
      LEAFGAP_$MMLC=0;
      OUTPUT ICM_$MMLC;
         (//'***ERROR IN CM ',I4,' ($MMLC)'/
         ' Inter-leaf air gap is negative - reset to 0   '//);
         IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
      ];

" IV. START POSITION OF LEAVES"
" ============================="
"START_$MMLC is calculated based on LEAFGAP and LEAFWIDTH"
"The leaves are distributed symetricaly with respect"
"to the central axis."

START_$MMLC=(0.5+0.5*TOT_LEAF_$MMLC-1.0)*LEAFGAP_$MMLC+
0.5*TOT_LEAF_$MMLC*LEAFWIDTH_$MMLC;
```

```
START_$MMLC=-START_$MMLC;

IF(ABS(START_$MMLC)-RMAX_CM(ICM_$MMLC)>1.E-5) [
  OUTPUT ICM_$MMLC;
   (//'***ERROR IN CM ',I4,' ($MMLC)'/
   ' START POSITION EXCEEDS CM BOUNDARY'//);
   IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
];

"V. FOCUS FOR DIVERGENT LEAF SIDES "
"================================== "
OUTPUT; (' Input the Z focus point of the leaf sides: ',$);
MINPUT ($MMLC) ZFOCUS_$MMLC(1); (F15.0);
OUTPUT ZFOCUS_$MMLC(1); (F12.5/);
IF(ABS(ZFOCUS_$MMLC(1)-ZMIN_$MMLC)<1.E-5) [
   ZFOCUS_$MMLC(1)=ZMIN_$MMLC-1.E-4;
   OUTPUT ICM_$MMLC,ZFOCUS_$MMLC(1);
      (//' ***ERROR IN CM ',I4,' ($MMLC)'/
       ' ZFOCUS(1) cannot be equal to ZMIN_$MMLC'/
       ' ZFOCUS(1) reset to ',F15.5,' cm for now'//);
   IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
 ]
ELSEIF(ZFOCUS_$MMLC(1)>ZMIN_$MMLC & ZFOCUS_$MMLC(1)<ZMAX_$MMLC) [
   ZFOCUS_$MMLC(1)=ZMAX_$MMLC;
   OUTPUT ICM_$MMLC,ZFOCUS_$MMLC(1);
      (//' ***ERROR IN CM ',I4,' ($MMLC)'/
       ' ZFOCUS(1) is between ZMIN_$MMLC and ZMAX_$MMLC'/
       ' This will cause leaf sides to overlap'/
       ' ZFOCUS(1) reset to ',F15.5,' cm for now'//);
   IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
 ];

"VI. OBTAIN PARAMETERS IN DIRECTION ALONG LEAF MOVEMENT"
"======================================================================"

OUTPUT; (' Input the leaves dimensions along leaf movement',$);
MINPUT ($MMLC) HEDGEUP_O_$MMLC, HEDGEDOWN_O_$MMLC, WEDGEUP_O_$MMLC,
               WEDGEDOWN_O_$MMLC, HEDGEUP_U_$MMLC, HEDGEDOWN_U_$MMLC,
               WEDGEUP_U_$MMLC, WEDGEDOWN_U_$MMLC; (15F15.0);
OUTPUT HEDGEUP_O_$MMLC, HEDGEDOWN_O_$MMLC, WEDGEUP_O_$MMLC,
               WEDGEDOWN_O_$MMLC, HEDGEUP_U_$MMLC, HEDGEDOWN_U_$MMLC,
               WEDGEUP_U_$MMLC, WEDGEDOWN_U_$MMLC; (15F15.5);


"VII. OBTAIN THE Y-AXIS COORDINATES "
"================================== "

OUTPUT; (' Input the leaves dimensions in direction perpendicular to
               leaf movement',$);
MINPUT ($MMLC) YGROOVEUP_U_$MMLC, WGROOVEDOWN_U_$MMLC, WGROOVEUP_U_$MMLC,
               YGROOVEDOWN_O_$MMLC, WGROOVEUP_O_$MMLC, WGROOVEDOWN_O_$MMLC; (15F15.0);
OUTPUT YGROOVEUP_U_$MMLC, WGROOVEDOWN_U_$MMLC, WGROOVEUP_U_$MMLC,
               YGROOVEDOWN_O_$MMLC, WGROOVEUP_O_$MMLC, WGROOVEDOWN_O_$MMLC; (15F15.5);

"VIII. OBTAIN THE Z-AXIS COORDINATES "
"================================== "

OUTPUT; (' Input the leaves dimensions along Z axis',$);
MINPUT ($MMLC) ZGROOVEDOWN_U_$MMLC, HGROOVEUP_U_$MMLC, HGROOVEDOWN_U_$MMLC,
               ZGROOVEUP_O_$MMLC, HGROOVEUP_O_$MMLC, HGROOVEDOWN_O_$MMLC; (15F15.0);
OUTPUT ZGROOVEDOWN_U_$MMLC, HGROOVEUP_U_$MMLC, HGROOVEDOWN_U_$MMLC,
               ZGROOVEUP_O_$MMLC, HGROOVEUP_O_$MMLC, HGROOVEDOWN_O_$MMLC; (15F15.5);

"An air region is leaved above and under the leaf material, in order to take"
"into account the arrangment of the leaves with respect to the FOCUS."
"Therefore the leaf itself is between ZREG_$MMLC(I,2) and ZREG_$MMLC(I,6)"

"IX. CALCULATE PARAMETERS IN Z-AXIS"
"================================== "

"An air region is leaved above and under the leaf material, in order to take"
"into account the arrangment of the leaves with respect to the FOCUS."
"Therefore the leaf itself is between ZREG_$MMLC(I,2) and ZREG_$MMLC(I,6)"

I=0;
L=0;
DO K=1,TOT_LEAF_$MMLC["Unten leaf"
  I=I+1;
```

```
  L=L+1;
  IF(MOD(L,2)~=0)["Unten leaf"
        ZREG_$MMLC(I,1)=ZMIN_$MMLC;
        ZREG_$MMLC(I,2)=ZREG_$MMLC(I,1)+0.5;
        ZREG_$MMLC(I,3)=ZREG_$MMLC(I,2)+HGROOVEUP_U_$MMLC;
        ZREG_$MMLC(I,4)=ZREG_$MMLC(I,2)+ZTHICK_$MMLC-
                                    (HGROOVEDOWN_U_$MMLC+ZGROOVEDOWN_U_$MMLC);
        ZREG_$MMLC(I,5)=ZREG_$MMLC(I,4)+HGROOVEDOWN_U_$MMLC;
        ZREG_$MMLC(I,6)=ZREG_$MMLC(1,5)+ZGROOVEDOWN_U_$MMLC;
        ZREG_$MMLC(I,7)=ZMAX_$MMLC;
  ]
  ELSE["Oben leaf"
    ZREG_$MMLC(I,1)=ZMIN_$MMLC;
        ZREG_$MMLC(I,2)=ZREG_$MMLC(I,1)+0.5;
        ZREG_$MMLC(I,3)=ZREG_$MMLC(I,2)+ZGROOVEUP_O_$MMLC;
        ZREG_$MMLC(I,4)=ZREG_$MMLC(I,3)+HGROOVEUP_O_$MMLC;
        ZREG_$MMLC(I,5)=ZREG_$MMLC(I,2)+ZTHICK_$MMLC-HGROOVEDOWN_O_$MMLC;
        ZREG_$MMLC(I,6)=ZREG_$MMLC(I,2)+ZTHICK_$MMLC;
        ZREG_$MMLC(I,7)=ZMAX_$MMLC;
  ]
];

"X. CALCULATE PARAMETERS IN Y-AXIS"
"================================== "

TEMP1 = (ZREG_$MMLC(1,2)-ZFOCUS_$MMLC(1));

"SURPARA1_$MMLC is defined respect to ZREG(I,2) (i.e. the upper"
"border of the leaves)."
"Both kind of leaves are focused at ZMIN, later the unten leaves"
"will be shifted 0.55 [cm] in Z direction."

I=0;
L=0;
IF(TOT_LEAF_$MMLC>0) [
   DO K = 1,TOT_LEAF_$MMLC [
     I=I+1;
     L=L+1;
     IF(I=1)[
       YREG_$MMLC(I,1)=START_$MMLC;
     ]
     ELSE["definition of left border of leaves"
         YREG_$MMLC(I,1)=YREG_$MMLC(I-1,6);
     ]
     "we have now defined YREG_$MMLC(I,1)"
     "the gap between the leaves is included in each leaf"
     "for MMLC there are two kind of leaves: Oben and Unten"

     IF(MOD(L,2)~=0)["Unten leaf"
               YREG_$MMLC(I,2)=YREG_$MMLC(I,1)+YGROOVEUP_U_$MMLC;
               YREG_$MMLC(I,3)=YREG_$MMLC(I,1)+WGROOVEDOWN_U_$MMLC;
               YREG_$MMLC(I,4)=YREG_$MMLC(I,2)+WGROOVEUP_U_$MMLC;
               YREG_$MMLC(I,5)=YREG_$MMLC(I,1)+LEAFWIDTH_$MMLC;
               YREG_$MMLC(I,6)=YREG_$MMLC(I,5)+LEAFGAP_$MMLC;
     ]
     ELSE["Oben leaf"
               YREG_$MMLC(I,2)=YREG_$MMLC(I,1)+YGROOVEDOWN_O_$MMLC;
               YREG_$MMLC(I,3)=YREG_$MMLC(I,1)+WGROOVEUP_O_$MMLC;
               YREG_$MMLC(I,4)=YREG_$MMLC(I,2)+WGROOVEDOWN_O_$MMLC;
               YREG_$MMLC(I,5)=YREG_$MMLC(I,1)+LEAFWIDTH_$MMLC;
               YREG_$MMLC(I,6)=YREG_$MMLC(I,5)+LEAFGAP_$MMLC;
     ]
     DO J = 1, 6 [
               SURPARA1_$MMLC(I,J)=YREG_$MMLC(I,J)/TEMP1;
        ]; "End of J Loop"
  ]; "End of K Loop"
];   "End of IF statement"


"X. FOCUSED ARRAGEMENT OF THE LEAVES AND TILTING"
"============================================="

"Now starts the arrangement of the leaves. Basically it is a rotation with"
"respect to the FOCUS. The angles of rotation are obtained using"
"SURPARA1_$MMLC."
"Finally the new planes defining this boundaries are calculated."

"Determination of the angle of rotation for each leaf"
```

```
DO I=1,TOT_LEAF_$MMLC[
  BETA_$MMLC(I)=0.5*(ATAN(SURPARA1_$MMLC(I,1))+ATAN(SURPARA1_$MMLC(I,6)));
];

"For the focusing only the planes in Z (i.e. Z=ZREG(I,J)) are rotated"
"The solution is the equation Z=MBETA*Y+NBETA, when Z=ZREG(I,J) is"
"rotated about ZFOCUS in an angle BETA using the known rotation matrix"

DO I=1,TOT_LEAF_$MMLC[

"In order to maintain the borders clear, the upper and lower surfaces"
"in Z are constrained to be ZMIN and ZMAX, as defined above."

  MBETA_$MMLC(I,1)=0.0;
  NBETA_$MMLC(I,1)=ZREG_$MMLC(I,1);
  MBETA_$MMLC(I,7)=0.0;
  NBETA_$MMLC(I,7)=ZREG_$MMLC(I,7);

  DO J=2,6[
       MBETA_$MMLC(I,J)=-TAN(BETA_$MMLC(I));
       NBETA_$MMLC(I,J)=ZREG_$MMLC(I,J)+(COS(BETA_$MMLC(I))-1)*ZFOCUS_$MMLC(1);
       NBETA_$MMLC(I,J)=NBETA_$MMLC(I,J)/COS(BETA_$MMLC(I));
  ];
];

"The Unten leaves are shifted radially 0.55 [cm] w/r to ZFOCUS"
"Therefore, only the N parameter changes for the surfaces"

DO I=1,TOT_LEAF_$MMLC[
       IF(MOD(I,2)~=0)["Unten"
              DO J=2,6[
                     NBETA_$MMLC(I,J)=NBETA_$MMLC(I,J)+0.55*(COS(BETA_$MMLC(I))-
                            MBETA_$MMLC(I,J)*SIN(BETA_$MMLC(I)));
              ]
       ]
]

"Now starts the tilting process. Is a rotation of the planes with respect"
"to ZTILT in an given angle TILT_$MMLC"

"Definition of variables needed (angle and point of tilting)"

OUTPUT; (' Input the Z tilting point of the leaves and tilting angle: ',$);
MINPUT ($MMLC) ZTILT_$MMLC, TILT_$MMLC; (15F15.0);
OUTPUT ZTILT_$MMLC, TILT_$MMLC; (15F15.5);

TILT_$MMLC=TILT_$MMLC*3.14159/180;

"First we start with the planes on Y. The result of rotating this planes"
"with form Y=SURPARA1*(Z-ZFOCUS) is Y=MYREG*(Z-ZTILT)+NYREG, where the"
"factor (Z-ZTILT) is maintained for sake of simplicity."

DO I=1,TOT_LEAF_$MMLC[
  DO J=1,6["One set of indexes for each Y boundary"
       MYREG_$MMLC(I,J)=COS(TILT_$MMLC)*SURPARA1_$MMLC(I,J)+SIN(TILT_$MMLC);
       MYREG_$MMLC(I,J)=MYREG_$MMLC(I,J)/(-SIN(TILT_$MMLC)*SURPARA1_$MMLC(I,J)+
                                   COS(TILT_$MMLC));
       NYREG_$MMLC(I,J)=SURPARA1_$MMLC(I,J)*(ZTILT_$MMLC-ZFOCUS_$MMLC(1));
  ];
];

"Finally, we finish with the planes on Z. The result of rotating this planes"
"is Z=MZREG*Y+NZREG"

DO I=1,TOT_LEAF_$MMLC[

"In order to maintain the borders clear, the upper and lower surfaces"
"in Z are constrained to be ZMIN and ZMAX, as defined above."

  MZREG_$MMLC(I,1)=0.0;
  NZREG_$MMLC(I,1)=ZREG_$MMLC(I,1);
  MZREG_$MMLC(I,7)=0.0;
  NZREG_$MMLC(I,7)=ZREG_$MMLC(I,7);

  DO J=2,6[
       MZREG_$MMLC(I,J)=-SIN(TILT_$MMLC)+MBETA_$MMLC(I,J)*COS(TILT_$MMLC);
       MZREG_$MMLC(I,J)=MZREG_$MMLC(I,J)/(COS(TILT_$MMLC)+MBETA_$MMLC(I,J)*
                                   SIN(TILT_$MMLC));
```

```
        NZREG_$MMLC(I,J)=NBETA_$MMLC(I,J)-ZTILT_$MMLC;
        NZREG_$MMLC(I,J)=NZREG_$MMLC(I,J)/(COS(TILT_$MMLC)+MBETA_$MMLC(I,J)*
                                    SIN(TILT_$MMLC))+ZTILT_$MMLC;
  ];
];

" Check to see if the MMLC will go outside the CM boundary "
TEMP1 = ABS(MYREG_$MMLC(1,1)*(ZMAX_$MMLC-ZTILT_$MMLC)+NYREG_$MMLC(1,1));
TEMP2 = ABS(MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(ZMAX_$MMLC-ZTILT_$MMLC)+
                      NYREG_$MMLC(TOT_LEAF_$MMLC,6));

IF(TEMP1>RMAX_CM(ICM_$MMLC)|TEMP2>RMAX_CM(ICM_$MMLC)) [
   OUTPUT; (/' The MMLC is not contained within the CM boundaries', /
        ' Please check ZFOCUS_$MMLC(1) and RMAX_CM(ICM_$MMLC)'/);
   STOP;
];

"IX. INPUT THE COORDINATES NEG_$MMLC AND POS_$MMLC "
"================================================== "
mlc_unit=70;
DO I=1, TOT_LEAF_$MMLC
[
    LEAFB_$MMLC(I)=0.0;LEAFA_$MMLC(I)=0.0;
]
IF(MODE_$MMLC=1|MODE_$MMLC=2)["dynamic or step-and-shoot leaf inputs"
  OUTPUT;(/' Input full name of file containing leaf opening data:'/);
  READ(i_input,'(A80)')mlc_file;
  OUTPUT mlc_file;(A80/);
  mlc_unit=egs_get_unit(mlc_unit);
  IF(mlc_unit<1)[
    $egs_fatal(*,
         'MLC leaf opening file: failed to get a free Fortran I/O unit');
  ]
  open(mlc_unit,file=mlc_file,status='old',err=:no-mlc-data-file:);
  read(mlc_unit,'(A80)') MLC_TITLE;
  read(mlc_unit,'(I10)') NFIELDS_$MMLC;
  OUTPUT NFIELDS_$MMLC; (I10);
  DO I=1,NFIELDS_$MMLC[
  "read in non-default leaf positions for each field"
     read(mlc_unit,'(F15.0)')INDEX_$MMLC(I);
     J=1;"counter for total no. of leaves input so far"
     LOOP[
       read(mlc_unit,'(2F15.0,I5)')NEG_$MMLC, POS_$MMLC, NUM_$MMLC;
       IF(NUM_$MMLC<=0) NUM_$MMLC=1;
       DO K=J,J+NUM_$MMLC-1[
         IF(K>TOT_LEAF_$MMLC) EXIT;
         LEAFNEG_$MMLC(K+(I-1)*TOT_LEAF_$MMLC)=NEG_$MMLC;
         LEAFPOS_$MMLC(K+(I-1)*TOT_LEAF_$MMLC)=POS_$MMLC;
       ]
       J=K;
     ]WHILE(J<=TOT_LEAF_$MMLC);
  ]
  CLOSE(UNIT=mlc_unit);

  OUTPUT;('******MLC SEQUENCE FILE READ******');
  IF (MODE_$MMLC=1) [
      OUTPUT; ('**dynamic delivery**');
  ]
  ELSEIF (MODE_$MMLC=2) [
      OUTPUT; ('**step and shoot delivery**');
  ]
]
ELSE[
OUTPUT; (' Input for MLC A and B leaf tips '/);
IF(ORIENT_$MMLC=1)[
       OUTPUT;(' Input min. X, max. X of top of opening in leaves,');
]
ELSE[
    OUTPUT;(' Input min. Y, max. Y of top of opening in leaves,');
]

OUTPUT;(' # of adjacent leaves with these coordinates:');
I=1;
LOOP[
  OUTPUT I;('   For leaf',I4,' :',$)
  MINPUT ($MMLC) NEG_$MMLC, POS_$MMLC, NUM_$MMLC; (2F15.0,I5);
  IF(NUM_$MMLC<=0) NUM_$MMLC=1;
  OUTPUT NEG_$MMLC, POS_$MMLC, NUM_$MMLC; (2F12.5,I5);
```

```
  IF(NEG_$MMLC > POS_$MMLC)[
     NEG_$MMLC = POS_$MMLC;
     OUTPUT ICM_$MMLC,I,I+NUM_$MMLC-1,NEG_$MMLC;
      (//' ***ERROR IN CM ',I4,' ($MMLC)'/
   ' Min. and max. opening coordinates in leaves ',I4,' - ',I4,' overlap'/
   ' Both coordinates set to ',F15.5,' cm for now'//);
     IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
                              ]

  IF(ABS(NEG_$MMLC)>RMAX_CM(ICM_$MMLC)|
     ABS(POS_$MMLC)>RMAX_CM(ICM_$MMLC)) [

     OUTPUT ICM_$MMLC,I,I+NUM_$MMLC-1;
      (//' ***ERROR IN CM ',I4,' ($MMLC)'/
      ' Tip of leaves ',I4,' - ',I4,' are outside CM '//);

     IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
                                             ];
  DO J=I,I+NUM_$MMLC-1["define opening for all leaves in group"
    IF(J>TOT_LEAF_$MMLC) EXIT;
    LEAFB_$MMLC(J)=NEG_$MMLC;
    LEAFA_$MMLC(J)=POS_$MMLC;
  ]
  I=J;
]WHILE(I<=TOT_LEAF_$MMLC);"End of Coordinate inputs for || direction "
]


"X. ESTABLISH TOP OF FIRST CM
"==============================

ZFRONT_$MMLC = Z_min_CM(ICM_$MMLC);

"
"XI. ESTABLISH START OF NEXT CM
"================================

Z_min_CM(ICM_$MMLC+1) = ZMAX_$MMLC;

"
"XII. GET ECUT, PCUT, DOSE SCORING ZONE AND MATERIAL IN EACH REGION
"=================================================================

IRA = IRSTART_$MMLC-1;
DO IR_$MMLC = 1,N_$MMLC ["loop through regions to get information"
   IRA = IRA+1;
   IF(IR_$MMLC=1) [
       OUTPUT IR_$MMLC;
        (/' Region',I4,' (MLC opening):'/
          ' ECUT, PCUT (MeV), DOSE ZONE (0=NO DOSE SCORED), IREGION_TO_BIT'/
          ' :',$);
   ]
   ELSEIF(IR_$MMLC=2) [
       OUTPUT IR_$MMLC;
        (/' Region',I4,' (MLC leaves):'/
' ECUT, PCUT (MeV), DOSE ZONE (0=NO DOSE SCORED), IREGION_TO_BIT, IGNOREGAPS'/
         ' :',$);
   ]
   IF(IR_$MMLC~=2)[
      MINPUT ($MMLC) ECUT(IRA),PCUT(IRA),DOSE_ZONE(IRA),IREGION_TO_BIT(IRA);
      (2F15.0,2I5);
      OUTPUT ECUT(IRA),PCUT(IRA),DOSE_ZONE(IRA),IREGION_TO_BIT(IRA);
             (2F15.5,2I5);
   ]
   ELSE[
      MINPUT ($MMLC) ECUT(IRA),PCUT(IRA),DOSE_ZONE(IRA),
                     IREGION_TO_BIT(IRA),IGNOREGAPS_$MMLC; (2F15.0,3I5);
      OUTPUT ECUT(IRA),PCUT(IRA),DOSE_ZONE(IRA),IREGION_TO_BIT(IRA),
             IGNOREGAPS_$MMLC; (2F15.5,3I5);
   ]
   IF(ECUT(IRA) < ECUTIN) [ECUT(IRA)=ECUTIN;];
   IF(PCUT(IRA) < PCUTIN) [PCUT(IRA)=PCUTIN;];
   OUTPUT IR_$MMLC; (' material of region ',I3,' ',$);
     $MED_INPUT($MMLC); " inputs character array MED_IN from unit 5, loops"
  "through array MEDIA(24,I) to check if medium was previously input."
  "If so, sets MED_INDEX to index of previous medium.  If not,"
  "increments NMED and sets MED_INDEX to NMED."
   MED(IRA) = MED_INDEX; " medium of the planar slab"
```

```
 ] "end of loop over IR_$MMLC"
;

IF(IGNOREGAPS_$MMLC=1 & IREJCT_GLOBAL>0)[
   IF(ORIENT_$MMLC=1)["leaves parallel to X"
       OUTPUT;(/' *******Range rejection in $MMLC will ignore all'/
               ' air gaps if the particle is in the leaves and has'/
               ' X < min. X of leaf openings (not including leaf ends)'/
               ' or X > max. X of leaf openings (not including ends)'/);
   ]
   ELSE["leaves parallel to Y"
       OUTPUT;(/' *******Range rejection in $MMLC will ignore all'/
               ' air gaps if the particle is in the leaves and has'/
               ' Y < min. Y of leaf openings (not including leaf ends)'/
               ' or Y > max. Y of leaf openings (not including ends)'/);
   ]
   IF(MODE_$MMLC=0)[
   DO I=1,TOT_LEAF_$MMLC["loop through leaves to find index of those with"
                         "max. +ve opening and min. -ve opening"
       IF(I=1)[
           MIN_INDEX=I;
           MAX_INDEX=I;
       ]
       ELSE[
           IF(LEAFB_$MMLC(I)<LEAFB_$MMLC(I-1))MIN_INDEX=I;
           IF(LEAFA_$MMLC(I)>LEAFA_$MMLC(I-1))MAX_INDEX=I;
       ]
   ]

   MIN_PLANE_$MMLC=LEAFB_$MMLC(MIN_INDEX);
   MAX_PLANE_$MMLC=LEAFA_$MMLC(MAX_INDEX);

   ]
]
ELSE["set to default"
   IGNOREGAPS_$MMLC=0;
]
"
"XIII. SET UP AIR GAP TO PREVIOUS CM IF PRESENT
"  ========================================
"
"  The air gap has the highest region number in the CM, even though its
"  the top of the component module.  This is to allow the assignment of
"  region numbers on input of the parameters of each local region
"  (mainly to assign the medium number of the region).
"  The air gap is then assigned after all of the
"  CM parameters have been input.
"
"note that if this is the first CM (ICM_$mlc=1) then the gap thickness
"Z_gap_THICK(ICM_$MMLC) = 0, which is used as a flag for no air gap

Z_gap_THICK(ICM_$MMLC) = ZMIN_$MMLC - Z_min_CM(ICM_$MMLC);
IF (Z_gap_THICK(ICM_$MMLC) < = 0.0) [
   Z_gap_THICK(ICM_$MMLC) = 0.;
   N_GAP_$MMLC = 0; "no air gap for this CM"
  ]
ELSE [
   N_GAP_$MMLC = 1; "this CM has an air gap"
   IRA = IRSTART_$MMLC+N_$MMLC; "absolute region number of air gap"
   MED(IRA) = AIR_INDEX; "medium is air"
 ];
"
"XIV. SET UP REGION NUMBERS
"==========================
"
"  This CM has N_$MMLC+N_GAP_$mlc regions
"
;
"Index last region
IREND_$MMLC = (IRSTART_$MMLC -1) + N_$MMLC+N_GAP_$MMLC;
NREG = NREG+N_$MMLC+N_GAP_$MMLC;
                            "Total no of regions in full geometry up
                            "to and including this CM
IF (NREG <= $MXREG) [
   IR_start_CM(ICM_$MMLC+1) = IREND_$MMLC+1;
  ]    "have not exceeded maximum region number
       "Index of first region in next CM:"
ELSE [
```

```
   OUTPUT ICM_$MMLC,NREG,$MXREG;
   (//' ***ERROR IN CM ',I4,' ($MMLC):'/
   T2,I4,' regions requested, only ',I4,' available'//);
   IERR_GEOM(ICM_$MMLC)=IERR_GEOM(ICM_$MMLC)+1;
 ];

"
"XV.  ESTABLISH CM BOUNDARY
"============================
"
RMAX_CM_FLAG(ICM_$MMLC) = 2; "put a square boundary about CM
"
"XVI. ESTABLISH DOSE SCORING ZONES AND BIT SETTING FOR EACH REGION
"=================================================================
"
IRA = IRSTART_$MMLC-1; "absolute region number"
DO IR_$MMLC=1,N_$MMLC ["loop over local region number"
   IRA = IRA+1;
   "dose-scoring zones"
   NDOSE_ZONE = MAX(DOSE_ZONE(IRA),NDOSE_ZONE); "Number of dose zones"
   MAX_BIT = MAX(IREGION_TO_BIT(IRA),MAX_BIT); " current maximum"
   "charged particle range rejection parameters"
   ESAVE(IRA)=ESAVE_GLOBAL; "Particles with total energies below ESAVE are"
                            "considered for range rejection"
   ECUTRR(IRA)=ECUT(IRA); "Minimum energy on exit from region GXD"
   E_min_out(ICM_$MMLC)=ECUT(IRA); "Minimum energy on exit from CM"
  ] "end of loop over IR_$MMLC

"XVII. ESTABLISH SUB-REGION IR VALUES
"====================================
"There are 3 X regions, 5 Y regions and 6 Z regions"
DO L=1,TOT_LEAF_$MMLC [
      DO I=1,3 [ "NX"
           DO J =1,5 [ "NY"
                DO K = 1,6 [ "NZ"
                  SUBINDEX_$MMLC(L,I,J,K)=1; "set all regions to air"
                  ];
            ];
         ];
      ];
;
"Now define regions containing leaf medium"
DO L=1,TOT_LEAF_$MMLC[
IF(MOD(L,2)~=0)["Unten"
  DO I =2,3 [
    DO K = 1,4 [
      SUBINDEX_$MMLC(L,I,K,3)=2;
      SUBINDEX_$MMLC(L,I,K,5)=2;
    ];
    DO K = 3,4 [
      SUBINDEX_$MMLC(L,I,K,4)=2;
    ];
    SUBINDEX_$MMLC(L,I,1,2)=2;
    SUBINDEX_$MMLC(L,I,4,2)=2;
  ];
]
ELSE["Oben"
  DO I =2,3 [
    DO K = 1,4 [
      SUBINDEX_$MMLC(L,I,K,2)=2;
      SUBINDEX_$MMLC(L,I,K,4)=2;
    ];
    DO K = 3,4 [
      SUBINDEX_$MMLC(L,I,K,3)=2;
    ];
    SUBINDEX_$MMLC(L,I,1,5)=2;
    SUBINDEX_$MMLC(L,I,4,5)=2;
  ];
]
];
  "   ======================
"
RETURN;

"XVIII. ERROR MESSAGES
"==================
"
:EOF_$MMLC:
```

```
;OUTPUT ICM;
  (//' *** ERROR *** unexpected end of file reading input for CM',I3);
STOP;

:ERROR_$MMLC:
;OUTPUT ICM;(//' *** ERROR *** format error on input for CM',I3);
STOP;

:no-mlc-data-file:
;OUTPUT;(//' *** ERROR: MLC leaf data file could not be opened');
STOP;

END;  "End of INPUT_$MMLC"

%E "Start of subroutine ISUMRY_$MMLC "
"***********************************************************************
"
"                           Subroutine ISUMRY_$MMLC
"                           ***********************
"
" Summarize input, write graphics file for EGS_Windows, and set parameters
" that require medium information obtained from HATCH call.
"
"***********************************************************************

;SUBROUTINE ISUMRY_$MMLC;

;IMPLICIT NONE;
;COMIN/ BOUNDS,CMs,CM_$MMLC,GEOM,IO_INFO,MEDIA,MISC,SCORE,UPHIOT,USER/;
"T>
"T>*********************************
"T>TYPE DECLARATIONS FOR ISUMRY_$MMLC
"T>*********************************
"T>
INTEGER
   ICOLOUR, "T>colour of CM for EGS_Windows
"    ID,     already defined gf  T>index of dose scoring zone
   IRA,     "T>absolute region number
   I,J,     "T>DO loop index
   ISTART,IEND, "T>indices for outputting info"
   YY1(9),          "T> YY1 and YY2 index subregion along perpendicular..
   YY2(9),          "T> direction to MLC orientation
   YY3(9),          "T> specific for target leaf
   YY4(9),          "T>
   YY5(9),          "T> same for full leaf
   YY6(9),          "T>
   YY7(9),          "T> same for isocenter leaf
   YY8(9);         "T>

$REAL  VOL_$MMLC(3), "T> region volumes
       Zo,             "T> THe z-coordinate of the leaf center
       XoP,XoN,        "T> The center of the rounded leaves ends along leaf
       M1,M2,          "T> Variables related to partial(rounded) leaf volume
       THETASUB,       "T> Angle at leaf end subtended by Z boundaries of reg.
       TOTALVOL,       "T> Total volume of MLC CM
       ZSQUARE,ZCUBE,AREA,    "T> Variables related to partial leaf volumes
       TEMP,TEMP1,TEMP2,TEMP3,"T> Variables related to partial leaf volumes
       HOLD;            "T> to hold a TEMP value

"Volume calculation is temporaly removed"

"   Summarize geometrical information for this CM in listing file
"   ==============================================================
"
WRITE(IOUTLIST,110) ICM_$MMLC,TITLE_$MMLC;
WRITE(IOUTLIST,120) Z_min_CM(ICM_$MMLC), RMAX_CM(ICM_$MMLC);
IF(N_GAP_$MMLC~=0) [
   WRITE(IOUTLIST,124) Z_min_CM(ICM_$MMLC),
                       ZMIN_$MMLC-Z_min_CM(ICM_$MMLC);
  ];
WRITE(IOUTLIST,122) ZTHICK_$MMLC;
IF(ORIENT_$MMLC~=0) [
   WRITE(IOUTLIST,123) 'X';
]
ELSE[
   WRITE(IOUTLIST,123) 'Y';
]
WRITE(IOUTLIST,127) LEAFGAP_$MMLC;
```

```
WRITE(IOUTLIST,128) LEAFWIDTH_$MMLC;
IF(ORIENT_$MMLC~=0) [
      WRITE(IOUTLIST,129)'Y','X';
 ]
ELSE [
      WRITE(IOUTLIST,129)'X','Y';
]
ISTART=1;
IEND=2;
LOOP[
  IF(IEND>TOT_LEAF_$MMLC |
     LEAFB_$MMLC(IEND)~=LEAFB_$MMLC(ISTART)|
     LEAFA_$MMLC(IEND)~=LEAFA_$MMLC(ISTART))[
     WRITE(IOUTLIST,130)ISTART,IEND-1,YREG_$MMLC(ISTART,1),
                        YREG_$MMLC(IEND-1,5),LEAFB_$MMLC(ISTART),LEAFA_$MMLC(ISTART);
     ISTART=IEND;
  ]
  ELSE[
     IEND=IEND+1;
  ]
]WHILE(ISTART<=TOT_LEAF_$MMLC);
WRITE(IOUTLIST,131);
IRA=IRSTART_$MMLC-1;
DO IR_$MMLC=1,N_$MMLC [
   IRA = IRA+1;
   IF(IR_$MMLC=1) [
      IF (MED(IRA)=0) ["Medium is vacuum"
         WRITE(IOUTLIST,140) IR_$MMLC,'opening',ECUT(IRA),PCUT(IRA),
           ECUTRR(IRA),ESAVE(IRA),DOSE_ZONE(IRA),
           IREGION_TO_BIT(IRA),'V','a','c','u','u','m';
       ]
      ELSE ["Medium is not vacuum"
         WRITE(IOUTLIST,140) IR_$MMLC,'opening',ECUT(IRA),PCUT(IRA),
         ECUTRR(IRA),ESAVE(IRA),DOSE_ZONE(IRA),
         IREGION_TO_BIT(IRA),(MEDIA(J,MED(IRA)),J=1,9);
      ];
   ]
   ELSEIF(IR_$MMLC=2 | IR_$MMLC=3) [
      IF (MED(IRA)=0) ["Medium is vacuum"
         WRITE(IOUTLIST,140) IR_$MMLC,'leaves',ECUT(IRA),PCUT(IRA),
           ECUTRR(IRA),ESAVE(IRA),DOSE_ZONE(IRA),
           IREGION_TO_BIT(IRA),'V','a','c','u','u','m';
       ]
      ELSE ["Medium is not vacuum"
         WRITE(IOUTLIST,140) IR_$MMLC,'leaves',ECUT(IRA),PCUT(IRA),
         ECUTRR(IRA),ESAVE(IRA),DOSE_ZONE(IRA),
         IREGION_TO_BIT(IRA),(MEDIA(J,MED(IRA)),J=1,9)
       ]
   ];
];

IF(N_GAP_$MMLC~=0) [
   IRA=IRSTART_$MMLC+N_$MMLC;
   WRITE(IOUTLIST,140) IR_$MMLC,'airgap',ECUT(IRA),PCUT(IRA),
         ECUTRR(IRA),ESAVE(IRA),DOSE_ZONE(IRA),IREGION_TO_BIT(IRA),
         (MEDIA(J,MED(IRA)),J=1,9);
   WRITE(IOUTLIST,141)'at top'
 ];

IF(IGNOREGAPS_$MMLC=1 & IREJCT_GLOBAL>0)[
  IF(ORIENT_$MMLC=1)["leaves parallel to X"
   WRITE(IOUTLIST,'(/'' *******Range rejection in $MMLC will ignore all''/
                  '' air gaps if the particle is in the leaves and has''/
                  '' X < min. X of leaf openings (not including leaf ends)''/
                  '' or X > max. X of leaf openings (not including ends)''/)');
  ]
  ELSE["leaves parallel to Y"
   WRITE(IOUTLIST,'(/'' *******Range rejection in $MMLC will ignore all''/
                  '' air gaps if the particle is in the leaves and has''/
                  '' Y < min. Y of leaf openings (not including leaf ends)''/
                  '' or Y > max. Y of leaf openings (not including ends)''/)');
  ]
]

110 FORMAT(/'  Component module',I3,' is a multi-leaf collimator',
           /' (3 regions )',
           /'  ------------------------------------------------',
```

```
          /'------------------',
          //T5,'Title: ',68A1);
120 FORMAT(/T2,'$MMLC geometry parameters:',
          /T2,'------------------------',
          /T2,'Distance from front of CM from reference plane = ',T51,
          F15.5,' cm',
          /T2,'Radius of outer boundary of CM = ',T51,F15.5,' cm');
122 FORMAT(T2,'Thickness of collimator = ',T51,F15.5,' cm');
123 FORMAT(T2,'Leaves open parallel to the ',A1,' axis');
124 FORMAT(T2,'There is an airgap starting at Z = ',F8.3,' cm with',
             ' thickness ',F8.3,' cm');
127 FORMAT(T2,'Gap between adjacent leaves = ',T51,F15.5,' cm');
128 FORMAT(/T2,'Dimensions for leaves (all in cm):'/
T2,'        Leaf width (incl. tongue):',F13.5/);
129 FORMAT(/
T2,' Leaves                ',A1,' range             ',A1,
' coordinates of opening'/
T2,'                           spanned (cm)              NEG   (cm)   POS');
130 FORMAT(T2,I3,'-',I3,4X,F10.5,' - ',F10.5,5X,F10.5,5X,F10.5);
131 FORMAT(/T2,'$MMLC region parameters:',
          /T2,'--------------------------',
          /T2,'local  location  electron  photon',
              '  range-rejection   dose  bit  medium'
          /T2,'region            cutoff   cutoff',
              '   level     max    zone  set'
          /T2,'                   (MeV)    (MeV)',
              '   (MeV)    (MeV)');
140 FORMAT(T2,I3,4X,A7,F10.3,F9.3,F8.3,F9.3,I6,I5,3X,9A1);
141 FORMAT(T9,A7);

"   Output representation of this component module to file for EGS_Windows
"   ======================================================================

IF (IWATCH=4|IZLAST=2) [   "Graphics file requested"
   ;
    ]
; "End of graphics output"

200 FORMAT(' ',I1,A3,A1,10(F7.2,','));
RETURN;
END;

%E "Start of subroutine HOWNEAR_$MMLC "
"*********************************************************************"
"                                                                     "
"                          Subroutine HOWNEAR_$MMLC               "
"                          ****************************               "
"                                                                     "
" Calculates min. distance to nearest region boundary                  "
" Used to be HOWNEAR macro, but is now called from that macro.         "
"                                                                     "
"*********************************************************************"

;SUBROUTINE HOWNEAR_$MMLC(DIST_sngl);
;
$IMPLICIT-NONE;

COMIN/CMs,CM_$MMLC,STACK,USER,EGS-IO/;

$REAL   DIST_sngl; "T> min. distance to nearest region boundary"
DOUBLE PRECISION   XYL1,XYL2, "T> X(NP) and Y(NP)"
                   XoN, XoP, DIST,
                   UVL1,UVL2,
                   TEMP1,TEMP2,TEMP3,TEMP4,
                   TEMP5,TEMP6,HOLE, "T> temp. distance variables"
                               XL1, XL2, XL3, XR1, XR2, XR3, "T> distance variables"
                   XL,XR,Lo,Ro,Zo;          "T>temp distance variables"
$INTEGER I,J, "T> looping index"
         NZ, "T> index of subregion in Z direction"
         NY, "T> index of subregion perpendicular to leaf opening direction"
         NX, "T> index of subregion in direction of leaf opening"
         LEAFIS, "T> leaf no. where particle is located"
         I1,I2; "T> used to mark min. max. Z boundaries for calculating dist"


IR_$MMLC=IR(NP)-IRSTART_$MMLC+1;
IF(IR_$MMLC=3) [ "in the air gap at the top"
  DIST=MIN(Z(NP)-ZFRONT_$MMLC,ZMIN_$MMLC-Z(NP));
```

```
  ]

ELSE [
  IF(ORIENT_$MMLC=1) [ XYL1=Y(NP);XYL2=X(NP);UVL1=V(NP);UVL2=U(NP); ]
  ELSE [ XYL1=X(NP);XYL2=Y(NP);UVL1=U(NP);UVL2=V(NP);];

  IF(IGNOREGAPS_$MMLC=1 & XYL2 < MIN_PLANE_$MMLC &
  XYL1 > (MYREG_$MMLC(1,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(1,1)) &
  XYL1 < (MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(Z(NP)-ZTILT_$MMLC)+
  NYREG_$MMLC(TOT_LEAF_$MMLC,6)) & IR_$MMLC=2)[
   "particle within negative leaves, ignore air gaps for range rejection"

   "distance to most -ve leaf side..."
   TEMP1=MYREG_$MMLC(1,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(1,1);
   TEMP1=ABS((TEMP1-XYL1)/SQRT(1+MYREG_$MMLC(1,1)**2));

   "distance to most +ve leaf side..."
   TEMP2=MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(Z(NP)-ZTILT_$MMLC)+
        NYREG_$MMLC(TOT_LEAF_$MMLC,6);
   TEMP2=ABS((TEMP2-XYL1)/SQRT(1+MYREG_$MMLC(TOT_LEAF_$MMLC,6)**2));

   DIST=MIN(Z(NP)-ZMIN_$MMLC,ZMIN_$MMLC+ZTHICK_$MMLC-Z(NP),
            MIN_PLANE_$MMLC-XYL2,TEMP1,TEMP2);
  ]
  ELSEIF(IGNOREGAPS_$MMLC=1 & XYL2 > MAX_PLANE_$MMLC &
  XYL1 > MYREG_$MMLC(1,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(1,1) &
  XYL1 < (MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(Z(NP)-ZTILT_$MMLC)+
        NYREG_$MMLC(TOT_LEAF_$MMLC,6)) & IR_$MMLC=2)[
   "particle within positive leaves, ignore air gaps for range rejection"

   "distance to most -ve leaf side..."
   TEMP1=MYREG_$MMLC(1,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(1,1);
   TEMP1=ABS((TEMP1-XYL1)/SQRT(1+MYREG_$MMLC(1,1)**2));

   "distance to most +ve leaf side..."
   TEMP2=MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(Z(NP)-ZTILT_$MMLC)+
        NYREG_$MMLC(TOT_LEAF_$MMLC,6);
   TEMP2=ABS((TEMP2-XYL1)/SQRT(1+MYREG_$MMLC(TOT_LEAF_$MMLC,6)**2));

   DIST=MIN(Z(NP)-ZMIN_$MMLC,ZMIN_$MMLC+ZTHICK_$MMLC-Z(NP),
            XYL2-MAX_PLANE_$MMLC,TEMP1,TEMP2);
  ]
  ELSE["do not ignore air gaps for range rejection"

LEAFIS=0; "Determine which leaf we are in, I is index of leaf number"
DO I = 1,TOT_LEAF_$MMLC [
    TEMP1=MYREG_$MMLC(I,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(I,1);
    TEMP2=MYREG_$MMLC(I,6)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(I,6);
    IF(XYL1>=TEMP1 & TEMP2>=XYL1) [
       LEAFIS=I; EXIT;];
 ];

IF(LEAFIS~=0) [
  TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
  TEMP2=MYREG_$MMLC(LEAFIS,6)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,6);
  IF(XYL1<TEMP1)[
    NY=1;
  ]
  ELSEIF(XYL1>TEMP2)[
    NY=5;
  ]
  ELSE[

    DO J =1,5 [ "Determine which Y region we are in"
      TEMP1=MYREG_$MMLC(LEAFIS,J)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,J);
      TEMP2=MYREG_$MMLC(LEAFIS,J+1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,J+1);
      IF(TEMP1<=XYL1 & XYL1<=TEMP2) [NY=J; EXIT;]
      ];
  ]
  "Z planes now are rotated"
  TEMP1=MZREG_$MMLC(LEAFIS,1)*XYL1+NZREG_$MMLC(LEAFIS,1);
  TEMP2=MZREG_$MMLC(LEAFIS,7)*XYL1+NZREG_$MMLC(LEAFIS,7);

  IF(Z(NP)<TEMP1)[
    NZ=1;
  ]
  ELSEIF(Z(NP)>TEMP2)[
    NZ=6;
```

```
  ]
  ELSE[
    DO J = 1,6 [ "Determine which Z region we are in for leaf I"
    TEMP1=MZREG_$MMLC(LEAFIS,J)*XYL1+NZREG_$MMLC(LEAFIS,J);
    TEMP2=MZREG_$MMLC(LEAFIS,J+1)*XYL1+NZREG_$MMLC(LEAFIS,J+1);
   IF((TEMP1<=Z(NP))&(Z(NP)<=TEMP2))[
       NZ=J; EXIT;];
    ];
  ]


"Determine which X region we are in"

"The semi-focusing leaf end is introduced. The ZX plane is divided in three"
"subregions: the opposing leaves and the gap between them."
"First the Z coordinate of each border of the leaf end is obtained,"
"calculating the ortogonal proyection of the position of the particle on each"
"border of the leaf end in the YZ plane."
"Then the equations that define the border of the leaf end can be obtained."

IF(MOD(LEAFIS,2)~=0)["Unten"
        ZXREG_$MMLC(1)=MZREG_$MMLC(LEAFIS,2)**2*XYL1
                       +MZREG_$MMLC(LEAFIS,2)*XYL1+NZREG_$MMLC(LEAFIS,2);
        ZXREG_$MMLC(1)=ZXREG_$MMLC(1)/(1+MZREG_$MMLC(LEAFIS,2)**2);
        ZXREG_$MMLC(2)=ZXREG_$MMLC(1)+HEDGEUP_U_$MMLC*
                       COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
        ZXREG_$MMLC(3)=ZXREG_$MMLC(2)+(ZTHICK_$MMLC-(HEDGEUP_U_$MMLC+
                       HEDGEDOWN_U_$MMLC))*COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
        ZXREG_$MMLC(4)=ZXREG_$MMLC(1)+ZTHICK_$MMLC*
                       COS(ATAN(MZREG_$MMLC(LEAFIS,2)));

        "Calculation of line equations for up and down diagonal boundaries"
        "1: up ; 2: down"
        "A: positive ; B: Negative"

        MAREG_$MMLC(1)=-WEDGEUP_U_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
        NAREG_$MMLC(1)=LEAFA_$MMLC(LEAFIS)+WEDGEUP_U_$MMLC-
                       ZXREG_$MMLC(1)*MAREG_$MMLC(1);
        MAREG_$MMLC(2)=WEDGEDOWN_U_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
        NAREG_$MMLC(2)=LEAFA_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MAREG_$MMLC(2);

        MBREG_$MMLC(1)=WEDGEUP_U_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
        NBREG_$MMLC(1)=LEAFB_$MMLC(LEAFIS)-WEDGEUP_U_$MMLC-
                       ZXREG_$MMLC(1)*MBREG_$MMLC(1);
        MBREG_$MMLC(2)=-WEDGEDOWN_U_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
        NBREG_$MMLC(2)=LEAFB_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MBREG_$MMLC(2);
]
ELSE["Oben"
    ZXREG_$MMLC(1)=MZREG_$MMLC(LEAFIS,2)**2*XYL1
                       +MZREG_$MMLC(LEAFIS,2)*XYL1+NZREG_$MMLC(LEAFIS,2);
        ZXREG_$MMLC(1)=ZXREG_$MMLC(1)/(1+MZREG_$MMLC(LEAFIS,2)**2);
        ZXREG_$MMLC(2)=ZXREG_$MMLC(1)+HEDGEUP_O_$MMLC*
                       COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
        ZXREG_$MMLC(3)=ZXREG_$MMLC(2)+(ZTHICK_$MMLC-(HEDGEUP_O_$MMLC+
                       HEDGEDOWN_O_$MMLC))*COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
        ZXREG_$MMLC(4)=ZXREG_$MMLC(1)+ZTHICK_$MMLC*
                       COS(ATAN(MZREG_$MMLC(LEAFIS,2)));

        "Calculation of line equations for up and down diagonal boundaries"
        "1: up ; 2: down"
        "A: positive ; B: Negative"

        MAREG_$MMLC(1)=-WEDGEUP_O_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
        NAREG_$MMLC(1)=LEAFA_$MMLC(LEAFIS)+WEDGEUP_O_$MMLC-
                       ZXREG_$MMLC(1)*MAREG_$MMLC(1);
        MAREG_$MMLC(2)=WEDGEDOWN_O_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
        NAREG_$MMLC(2)=LEAFA_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MAREG_$MMLC(2);

        MBREG_$MMLC(1)=WEDGEUP_O_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
        NBREG_$MMLC(1)=LEAFB_$MMLC(LEAFIS)-WEDGEUP_O_$MMLC-
                       ZXREG_$MMLC(1)*MBREG_$MMLC(1);
        MBREG_$MMLC(2)=-WEDGEDOWN_O_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
        NBREG_$MMLC(2)=LEAFB_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MBREG_$MMLC(2);
]

        "X region discrimination"

        IF(Z(NP)<=ZXREG_$MMLC(2))[
```

```
                TEMP1=MBREG_$MMLC(1)*Z(NP)+NBREG_$MMLC(1);
                TEMP2=MAREG_$MMLC(1)*Z(NP)+NAREG_$MMLC(1);
                IF((XYL2>TEMP1 & XYL2<TEMP2) | (XYL2=TEMP1 & UVL2>=0.0) |
                   (XYL2=TEMP2 & UVL2<0.0)) [NX=1;]
                ELSEIF((XYL2<TEMP1) | (XYL2=TEMP1 & UVL2<0.0)) [NX=2;]
                ELSEIF((XYL2>TEMP2) | (XYL2=TEMP2 & UVL2>=0.0)) [NX=3;]
        ]
        ELSEIF(Z(NP)>ZXREG_$MMLC(2) & Z(NP)<=ZXREG_$MMLC(3))[
                TEMP1=LEAFB_$MMLC(LEAFIS);
                TEMP2=LEAFA_$MMLC(LEAFIS);
                IF((XYL2>TEMP1 & XYL2<TEMP2) | (XYL2=TEMP1 & UVL2>=0.0) |
                   (XYL2=TEMP2 & UVL2<0.0)) [NX=1;]
                ELSEIF((XYL2<TEMP1) | (XYL2=TEMP1 & UVL2<0.0)) [NX=2;]
                ELSEIF((XYL2>TEMP2) | (XYL2=TEMP2 & UVL2>=0.0)) [NX=3;]
        ]
        ELSEIF(Z(NP)>ZXREG_$MMLC(3))[
                TEMP1=MBREG_$MMLC(2)*Z(NP)+NBREG_$MMLC(2);
                TEMP2=MAREG_$MMLC(2)*Z(NP)+NAREG_$MMLC(2);
                IF((XYL2>TEMP1 & XYL2<TEMP2) | (XYL2=TEMP1 & UVL2>=0.0) |
                   (XYL2=TEMP2 & UVL2<0.0)) [NX=1;]
                ELSEIF((XYL2<TEMP1) | (XYL2=TEMP1 & UVL2<0.0)) [NX=2;]
                ELSEIF((XYL2>TEMP2) | (XYL2=TEMP2 & UVL2>=0.0)) [NX=3;]
        ];

"Discrimination with respect to kind of leaf"
"Set minimum and maximum Z boundaries"

IF(MOD(LEAFIS,2)~=0)["Unten"
        IF(NY=5 & (NZ>=2 & NZ<=5)) [I1=2; I2=6;]
    ELSE[I1 = NZ; I2 = NZ+1;]
]
ELSE["Oben"
        IF(NY=5 & (NZ>=2 & NZ<=5)) [I1=2; I2=6;]
    ELSE[I1 = NZ; I2 = NZ+1;]
]


        "Perpendicular distance to a line."
        TEMP1=MZREG_$MMLC(LEAFIS,I1)*XYL1+NZREG_$MMLC(LEAFIS,I1);
    TEMP2=MZREG_$MMLC(LEAFIS,I2)*XYL1+NZREG_$MMLC(LEAFIS,I2);

    TEMP1=ABS((TEMP1-Z(NP))/
          SQRT(1+MZREG_$MMLC(LEAFIS,I1)**2));
    TEMP2=ABS((TEMP2-Z(NP))/
          SQRT(1+MZREG_$MMLC(LEAFIS,I2)**2));
    "Calculates distance to closest Z boundary"
    DIST=MIN(TEMP1,TEMP2);

    "Again, discrimination with respect to kind of leaf"
    "Calculate distance to nearest boundary perp to leaf openings -->"
IF(MOD(LEAFIS,2)~=0)["Unten"
        IF(NZ=3 & NY<=4)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,5)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,5);
      I1=1;
      I2=5;
        ]
    ELSEIF(NZ=1)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,6)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,6);
      I1=1;
      I2=6;
        ]
        ELSEIF(NZ=2 & (NY=2 | NY=3))[
      TEMP1=MYREG_$MMLC(LEAFIS,2)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,2);
      TEMP2=MYREG_$MMLC(LEAFIS,4)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,4);
      I1=2;
      I2=4;
        ]
        ELSEIF(NZ=4 & (NY=1 | NY=2))[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,3)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,3);
      I1=1;
      I2=3;
        ]
        ELSEIF(NZ=4 & (NY=3 | NY=4))[
      TEMP1=MYREG_$MMLC(LEAFIS,3)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,3);
      TEMP2=MYREG_$MMLC(LEAFIS,5)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,5);
```

```
      I1=3;
      I2=5;
       ]
       ELSEIF(NZ=5 & NY<=4)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,5)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,5);
      I1=1;
      I2=5;
       ]
       ELSEIF(NZ=6)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,6)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,6);
      I1=1;
      I2=6;
       ]
       ELSE[
      TEMP1=MYREG_$MMLC(LEAFIS,NY)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,NY);
      TEMP2=MYREG_$MMLC(LEAFIS,NY+1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,NY+1);
      I1=NY;
      I2=NY+1;
       ]
]
ELSE["Oben"
    IF(NZ=1)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,6)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,6);
      I1=1;
      I2=6;
       ]
       ELSEIF(NZ=2 & NY<=4)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,5)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,5);
      I1=1;
      I2=5;
       ]
       ELSEIF(NZ=3 & NY<=2)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,3)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,3);
      I1=1;
      I2=3;
       ]
       ELSEIF(NZ=3 & (NY=3 | NY=4))[
      TEMP1=MYREG_$MMLC(LEAFIS,3)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,3);
      TEMP2=MYREG_$MMLC(LEAFIS,5)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,5);
      I1=3;
      I2=5;
       ]
       ELSEIF(NZ=4 & NY<=4)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,5)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,5);
      I1=1;
      I2=5;
       ]
       ELSEIF(NZ=5 & (NY=2 | NY=3))[
      TEMP1=MYREG_$MMLC(LEAFIS,2)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,2);
      TEMP2=MYREG_$MMLC(LEAFIS,4)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,4);
      I1=2;
      I2=4;
       ]
       ELSEIF(NZ=6)[
      TEMP1=MYREG_$MMLC(LEAFIS,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
      TEMP2=MYREG_$MMLC(LEAFIS,6)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,6);
      I1=1;
      I2=6;
       ]
       ELSE[
      TEMP1=MYREG_$MMLC(LEAFIS,NY)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,NY);
      TEMP2=MYREG_$MMLC(LEAFIS,NY+1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,NY+1);
      I1=NY;
      I2=NY+1;
       ]
]
       "Perpendicular distance to a line."
    TEMP1=ABS((TEMP1-XYL1)/
          SQRT(1+MYREG_$MMLC(LEAFIS,I1)**2));
    TEMP2=ABS((TEMP2-XYL1)/
          SQRT(1+MYREG_$MMLC(LEAFIS,I2)**2));
    DIST = MIN(DIST,TEMP1,TEMP2);
```

```
 "NOW CHECK DISTANCE IN X DIRECTION"
 "ZXREG: Z projection of particle in upper and lower boundary of leaf"
 "MAREG & NAREG: line equation of upper and lower boundary of leaf"

IF(NX=1)[
        IF(Z(NP)<ZXREG_$MMLC(2))[
                XL1=ABS(MBREG_$MMLC(1)*Z(NP)+NBREG_$MMLC(1)-XYL2);
                XL1=XL1/SQRT(1+MBREG_$MMLC(1)**2);
                XL2=SQRT((LEAFB_$MMLC(LEAFIS)-XYL2)**2+(ZXREG_$MMLC(2)-Z(NP))**2);

                TEMP1=MBREG_$MMLC(1)*(Z(NP)+MBREG_$MMLC(1)*(XYL2-NBREG_$MMLC(1)));
                TEMP1=TEMP1/(1+MBREG_$MMLC(1)**2)+NBREG_$MMLC(1);

                IF(TEMP1>LEAFB_$MMLC(LEAFIS))[
                        XL=XL2;
                ]
                ELSE[XL=MIN(XL1,XL2);]

                XR1=ABS(MAREG_$MMLC(1)*Z(NP)+NAREG_$MMLC(1)-XYL2);
                XR1=XR1/SQRT(1+MAREG_$MMLC(1)**2);
                XR2=SQRT((LEAFA_$MMLC(LEAFIS)-XYL2)**2+(ZXREG_$MMLC(2)-Z(NP))**2);

                TEMP1=MAREG_$MMLC(1)*(Z(NP)+MAREG_$MMLC(1)*(XYL2-NAREG_$MMLC(1)));
                TEMP1=TEMP1/(1+MAREG_$MMLC(1)**2)+NAREG_$MMLC(1);

                IF(TEMP1<LEAFA_$MMLC(LEAFIS))[
                        XR=XR2;
                ]
                ELSE[XR=MIN(XR1,XR2);]
        ]
        ELSEIF(Z(NP)>=ZXREG_$MMLC(2) & Z(NP)<=ZXREG_$MMLC(3))[
                XL=ABS(LEAFB_$MMLC(LEAFIS)-XYL2);
                XR=ABS(LEAFA_$MMLC(LEAFIS)-XYL2);
        ]
        ELSEIF(Z(NP)>ZXREG_$MMLC(3))[
                XL1=ABS(MBREG_$MMLC(2)*Z(NP)+NBREG_$MMLC(2)-XYL2);
                XL1=XL1/SQRT(1+MBREG_$MMLC(2)**2);
                XL2=SQRT((LEAFB_$MMLC(LEAFIS)-XYL2)**2+(ZXREG_$MMLC(3)-Z(NP))**2);

                TEMP1=MBREG_$MMLC(2)*(Z(NP)+MBREG_$MMLC(2)*(XYL2-NBREG_$MMLC(2)));
                TEMP1=TEMP1/(1+MBREG_$MMLC(2)**2)+NBREG_$MMLC(2);

                IF(TEMP1>LEAFB_$MMLC(LEAFIS))[
                        XL=XL2;
                ]
                ELSE[XL=MIN(XL1,XL2);]

                XR1=ABS(MAREG_$MMLC(2)*Z(NP)+NAREG_$MMLC(2)-XYL2);
                XR1=XR1/SQRT(1+MAREG_$MMLC(2)**2);
                XR2=SQRT((LEAFA_$MMLC(LEAFIS)-XYL2)**2+(ZXREG_$MMLC(3)-Z(NP))**2);

                TEMP1=MAREG_$MMLC(2)*(Z(NP)+MAREG_$MMLC(2)*(XYL2-NAREG_$MMLC(2)));
                TEMP1=TEMP1/(1+MAREG_$MMLC(2)**2)+NAREG_$MMLC(2);

                IF(TEMP1<LEAFA_$MMLC(LEAFIS))[
                        XR=XR2;
                ]
                ELSE[XR=MIN(XR1,XR2);]
        ]
]
ELSEIF(NX=2)[
        XL=ABS(-RMAX_CM(ICM_$MMLC)-XYL2);

        XR1=ABS(MBREG_$MMLC(1)*Z(NP)+NBREG_$MMLC(1)-XYL2);
        XR1=XR1/SQRT(1+MBREG_$MMLC(1)**2);

        XR2=ABS(LEAFB_$MMLC(LEAFIS)-XYL2);

        XR3=ABS(MBREG_$MMLC(2)*Z(NP)+NBREG_$MMLC(2)-XYL2);
        XR3=XR3/SQRT(1+MBREG_$MMLC(2)**2);

        XR=MIN(XR1,XR2);
        XR=MIN(XR,XR3);
]
ELSEIF(NX=3)[
        XL1=ABS(MAREG_$MMLC(1)*Z(NP)+NAREG_$MMLC(1)-XYL2);
        XL1=XL1/SQRT(1+MAREG_$MMLC(1)**2);
```

```
        XL2=ABS(LEAFA_$MMLC(LEAFIS)-XYL2);

        XL3=ABS(MAREG_$MMLC(2)*Z(NP)+NAREG_$MMLC(2)-XYL2);
        XL3=XL3/SQRT(1+MAREG_$MMLC(2)**2);

        XR=ABS(RMAX_CM(ICM_$MMLC)-XYL2);

        XL=MIN(XL1,XL2);
        XL=MIN(XL,XL3);
]

 DIST=MIN(DIST,XL,XR);
 IF(DIST<0) [OUTPUT LEAFIS,NX,NY,NZ; ('negative dist: ',4I3);]
]
ELSE[ "beyond outer edges of leaf bank"
  IF(XYL1 <= MYREG_$MMLC(1,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(1,1))[
      TEMP1=MYREG_$MMLC(1,1)*(Z(NP)-ZTILT_$MMLC)+NYREG_$MMLC(1,1);
      TEMP1=ABS((TEMP1-XYL1)/SQRT(1+MYREG_$MMLC(1,1)**2));
    ]
  ELSEIF(XYL1 >= MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(Z(NP)-ZTILT_$MMLC)+
       NYREG_$MMLC(TOT_LEAF_$MMLC,6))[
      TEMP1=MYREG_$MMLC(TOT_LEAF_$MMLC,6)*(Z(NP)-ZTILT_$MMLC)+
               NYREG_$MMLC(TOT_LEAF_$MMLC,6);
      TEMP1=ABS((TEMP1-XYL1)/SQRT(1+MYREG_$MMLC(TOT_LEAF_$MMLC,6)**2));
    ]
  DIST=MIN(Z(NP)-ZMIN_$MMLC,ZMAX_$MMLC-Z(NP),TEMP1);
 ]
 ]
 ];
 DIST_sngl=SNGL(DIST);
RETURN;
END; "End of subroutine HOWNEAR_$MMLC"
"***********************************************************************"
"End of MMLC_cm.mortran"
```

C.2 MMLC_macros.mortran

```
%E
/*****************************************************************************
 *
 *  $Id: MMLC_macros.mortran $
 *
 *****************************************************************************/
"-------------------------------------------------------------------"
"     MMLC miscellaneous replacement macros                        "
"-------------------------------------------------------------------"
"                                                                   "
REPLACE {$MAX_N_$MMLC} WITH {{REDUCE $MAXIMUM_N_$MMLC}};
"       ==========="

REPLACE {$MAXIMUM_N_$MMLC} WITH {3};
"       ==================="

" THE MAX # OF the leaves TO BE ALLOWED IN THIS MODULE"
REPLACE {$MAXLEAF} WITH {160}

" The max no. of different fields--dynamic and step-and-shoot only"
REPLACE {$MAXFIELD_$MMLC} WITH {100}

"used for arrays that store data for each leaf for each field"
REPLACE {$MAXFIELDLEAF} WITH {{COMPUTE $MAXLEAF*$MAXFIELD_$MMLC}}
;
"-------------------------------------------------------------------"
"     MMLC component module common                                 "
"-------------------------------------------------------------------"

"V>COMMON/CM_$MMLC/
"V>================
"V>ICM_$MMLC        = index of CM, set as ICM in INPUT_$MMLC,not reset
"V>IRSTART_$MMLC    = first region number for this CM
"V>IREND_$MMLC      = last region number for this CM
"V>N_$MMLC          = number of regions in CM
"V>TITLE_$MMLC      = title of CM
"V>ZMAX_$MMLC       = back of MMLC
"V>ZTHICK_$MMLC     = Thickness of leaves
"V>ZFRONT_$MMLC     = Upstream Z boundary of this CM
"V>TOT_LEAF_$MMLC = total no. of leaves in MLC
"V>LEAFWIDTH_$MMLC= the width of each leaf in group I at ZMIN_$MMLC
"V>                       excluding the tongue
"V>ORIENT_$MMLC     = the index to indicate the leave direction
"V>                     0: default, leaf orientation in y
"V>                     1:                          in x
"V>START_$MMLC      = the start position wrt the CAX of the lowermost
"V>                       leaf ie leaf 1 tongue as projected to ZMIN_$MMLC
"V>LEAFGAP_$MMLC    = the width of the interleaf air gap at ZMIN_$MMLC
"V>ZREG_$MMLC(TOT_LEAF_$MMLC,8) = the z boundaries of the 7 sub-regions
"V>                                  in z direction
"V>YREG_$MMLC(TOT_LEAF_$MMLC,7)
"V>                   = the boundaries along the perpendicular direction
"V>                     to the leaf orientation of sub-regions
"V>SUBINDEX_$MMLC   = an index number to represent which region the
"V>          belongs in based on sub-dividing each leaf into regions.
"V>SURPARA1_$MMLC($MAXLEAF,6)  the parameters to describe the leaf side
"V>                             surface i.e. tangent along that side.
"V>IR_$MMLC          local region number
"V>LEAFWIDTH_$MMLC= width of leaf of type LEAFTYPE excl. tongue
"V> BETA_$MMLC($MAXLEAF)= position (angle) of each leaf of the bank with
"V>                                       respect to the beam central axis.
"V> MBETA_$MMLC($MAXLEAF,7), NBETA_$MMLC($MAXLEAF,7)=parameters of the plane
"V>                                       equations in Z direction used to made the focused
"V>                                       arrangement of the leaves.
"V> MZREG_$MMLC($MAXLEAF,7), NZREG_$MMLC($MAXLEAF,7), MYREG_$MMLC($MAXLEAF,7),
"V> NYREG_$MMLC($MAXLEAF,7)= parameters of the plane equations that define the
"V>                                       boundaries of each subregion in the plane
perpendicular
"V>                                       to the movement of the leaves.
"V> MAREG_$MMLC(2), NAREG_$MMLC(2), MBREG_$MMLC(2),
"V> NBREG_$MMLC(2)= parameters of the plane equations that define the
"V>                                       boundaries of each subregion in the plane along
"V>                                       the movement of the leaves.
"V> ZTILT_$MMLC, TILT_$MMLC= tilting point and tilting angle.
"V> *GROOVE,*EDGE_$MMLC= physical dimensions of the leaves.
;
```

```
REPLACE{;COMIN/CM_$MMLC/;} WITH {
;COMMON/CM_$MMLC/EPS,
      ZMAX_$MMLC,
      ZFRONT_$MMLC,
      START_$MMLC,
      LEAFGAP_$MMLC,
      SURPARA1_$MMLC,
      ZREG_$MMLC,
      YREG_$MMLC,
      BETA_$MMLC,
      MBETA_$MMLC,
      NBETA_$MMLC,
      MZREG_$MMLC,
      NZREG_$MMLC,
      ZXREG_$MMLC,
      MAREG_$MMLC,
      NAREG_$MMLC,
      MBREG_$MMLC,
      NBREG_$MMLC,
      MYREG_$MMLC,
      NYREG_$MMLC,
      ZTILT_$MMLC,
      TILT_$MMLC,
      LEAFWIDTH_$MMLC,
      ICM_$MMLC,IRSTART_$MMLC,IREND_$MMLC,N_$MMLC,N_GAP_$MMLC,
      IR_$MMLC, ORIENT_$MMLC,
      SUBINDEX_$MMLC,
      TITLE_$MMLC,
      ZGROOVEUP_O_$MMLC,
      HGROOVEUP_O_$MMLC,
      HGROOVEDOWN_O_$MMLC,
      YGROOVEDOWN_O_$MMLC,
      WGROOVEUP_O_$MMLC,
      WGROOVEDOWN_O_$MMLC,
      ZGROOVEDOWN_U_$MMLC,
      HGROOVEUP_U_$MMLC,
      HGROOVEDOWN_U_$MMLC,
      YGROOVEUP_U_$MMLC,
      WGROOVEUP_U_$MMLC,
      WGROOVEDOWN_U_$MMLC,
      HEDGEUP_O_$MMLC,
      HEDGEDOWN_O_$MMLC,
      WEDGEUP_O_$MMLC,
      WEDGEDOWN_O_$MMLC,
      HEDGEUP_U_$MMLC,
      HEDGEDOWN_U_$MMLC,
      WEDGEUP_U_$MMLC,
      WEDGEDOWN_U_$MMLC;
   DOUBLE PRECISION
      EPS,
      ZMAX_$MMLC,
      ZFRONT_$MMLC,
      START_$MMLC,
      LEAFGAP_$MMLC,
      SURPARA1_$MMLC($MAXLEAF,7),
      ZREG_$MMLC($MAXLEAF,10),
      YREG_$MMLC($MAXLEAF,7),
      BETA_$MMLC($MAXLEAF),
      MBETA_$MMLC($MAXLEAF,7),
      NBETA_$MMLC($MAXLEAF,7),
      MZREG_$MMLC($MAXLEAF,7),
      NZREG_$MMLC($MAXLEAF,7),
      MAREG_$MMLC(2),
      NAREG_$MMLC(2),
      MBREG_$MMLC(2),
      NBREG_$MMLC(2),
      MYREG_$MMLC($MAXLEAF,6),
      NYREG_$MMLC($MAXLEAF,6),
      ZTILT_$MMLC,
      TILT_$MMLC,
      ZXREG_$MMLC(4),
      ZGROOVEUP_O_$MMLC,
      HGROOVEUP_O_$MMLC,
      HGROOVEDOWN_O_$MMLC,
      YGROOVEDOWN_O_$MMLC,
      WGROOVEUP_O_$MMLC,
      WGROOVEDOWN_O_$MMLC,
      ZGROOVEDOWN_U_$MMLC,
```

```
      HGROOVEUP_U_$MMLC,
      HGROOVEDOWN_U_$MMLC,
      YGROOVEUP_U_$MMLC,
      WGROOVEUP_U_$MMLC,
      WGROOVEDOWN_U_$MMLC,
      HEDGEUP_O_$MMLC,
      HEDGEDOWN_O_$MMLC,
      WEDGEUP_O_$MMLC,
      WEDGEDOWN_O_$MMLC,
      HEDGEUP_U_$MMLC,
      HEDGEDOWN_U_$MMLC,
      WEDGEUP_U_$MMLC,
      WEDGEDOWN_U_$MMLC;
  $REAL
      LEAFWIDTH_$MMLC;
  INTEGER
      ICM_$MMLC,IRSTART_$MMLC,IREND_$MMLC,N_$MMLC,N_GAP_$MMLC,
      IR_$MMLC, ORIENT_$MMLC,
      SUBINDEX_$MMLC($MAXLEAF,3,5,6);
  CHARACTER*1 TITLE_$MMLC(60);

}
 "end of replacement defining common for this CM"

"V>COMMON/USERMMLC/
"V>
"V> the following variables are required in the the main beam code
"V> to set leaf positions during the simulation when dynamic and step-and-shoot
"V> (MODE_$MMLC=1,2) options are used.  This common block is part of
"V> the larger USER common block.  It is defined as {;} in
"V> beamnrc_user_macros.mortran, but is superceded by this definition if
"V> MMLC is present in the accelerator.
"V>
"V>LEAFA_$MMLC($MAXLEAF) = coordinates of A (+ve) side leaves at
"V>                     ZMIN_$MMLC
"V>LEAFB_$MMLC($MAXLEAF) = coordinates of B (-ve) side leaves at
"V>                     ZMIN_$MMLC
"V>SURPARA2_B_$MMLC($MAXLEAF)the parameters to describe the leaf end
"V>                              surface i.e. tangent along that end.
"V>SURPARA2_A_$MMLC($MAXLEAF)the parameters to describe the leaf end
"V>                              surface i.e. tangent along that end.
"V>ZFOCUS_$MMLC(2) = the two focus point coordinates: 1. for leave side
"V>                                                   2. for leave end.
"V>ZMIN_$MMLC       = Front of MMLC
"V>ZTHICK_$MMLC     = Thickness of leaves
"V>MIN_PLANE_$MMLC  min. plane perp. to leaf direction.  For particles
"V>                    in leaves with position < MIN_PLANE_$MMLC,
"V>                    air gaps and driving screw holes will be ignored when
"V>                    doing range rejection (IGNOREGAPS_$MMLC=1)
"V>MAX_PLANE_$MMLC  max. plane perp. to leaf direction.  For particles
"V>                    in leaves with position > MAX_PLANE_$MMLC,
"V>                    air gaps and driving screw holes will be ignored when
"V>                    doing range rejection (IGNOREGAPS_$MMLC=1)
"V>LEAFNEG_$MMLC($MAXFIELDLEAF) = negative opening coordinates for all leaves
"V>                                  for all fields.
"V>                                  LEAFNEG_$MMLC(J+(I-1)*TOT_LEAF_$MMLC)
"V>                                  defines coordinate for leaf J in field I
"V>LEAFPOS_$MMLC($MAXFIELDLEAF) = positive opening coordinates for all leaves
"V>                                  for all fields.
"V>INDEX_$MMLC($MAXFIELD) = index for field I.  If INDEX_$MMLC(I)>=
"V>                           RNDM1_$MMLC then field I is used.
"V>INDEX1_$MMLC,INDEX2_$MMLC = temporary variables to store
"V>                                  INDEX_$MMLC(I) values so that
"V>                                  leaf positions can be interpolated during
"V>                                  dynamic simulations
"V>POS1_$MMLC,POS2_$MMLC,NEG1_$MMLC,NEG2_$MMLC = temporary
"V>                                variables to store LEAFNEG_$MMLC and
"V>                                LEAFPOS_$MMLC to allow leaf positions
"V>                                to be interpolated during dynamic field
"V>                                simulations
"V>RNDM1_$MMLC = random no. (0,1) selected before each history and compared
"V>                  to INDEX_$MMLC(I) to determin what field to use
"V>NFIELDS_$MMLC = no. of fields
"V>MODE_$MMLC = 0 for static field
"V>               1 for dynamic field
"V>               2 for step-and-shoot
"V>I_$MMLC,J_$MMLC = looping indices
"V>IGNOREGAPS_$MMLC Set to 1 to ignore air gaps and driving screw holes
```

```
"I>                    when doing range rejection
"V>                    for particles in the leaves and beyond the most open
"V>                    leaf.  0 (default) otherwise.
"V>MININD_$MMLC,MAXIND_$MMLC = indices used to determine
"V>                                 MIN_PLANE_$MMLC and MAX_PLANE_$MMLC

REPLACE {;COMIN/USER-MMLC/;} WITH {;
COMMON/USERMMLC/LEAFA_$MMLC($MAXLEAF),
LEAFB_$MMLC($MAXLEAF),SURPARA2_B_$MMLC($MAXLEAF),
SURPARA2_A_$MMLC($MAXLEAF),ZMIN_$MMLC,
ZFOCUS_$MMLC(2),ZTHICK_$MMLC,
MIN_PLANE_$MMLC,MAX_PLANE_$MMLC,
LEAFNEG_$MMLC($MAXFIELDLEAF),
LEAFPOS_$MMLC($MAXFIELDLEAF),INDEX_$MMLC($MAXFIELD_$MMLC),
INDEX1_$MMLC,INDEX2_$MMLC,POS1_$MMLC,POS2_$MMLC,
NEG1_$MMLC,NEG2_$MMLC,RNDM1_$MMLC,
NFIELDS_$MMLC,MODE_$MMLC,
J_$MMLC,I_$MMLC,TOT_LEAF_$MMLC,IGNOREGAPS_$MMLC,
MININD_$MMLC,MAXIND_$MMLC;
DOUBLE PRECISION LEAFA_$MMLC,LEAFB_$MMLC,
       SURPARA2_B_$MMLC,SURPARA2_A_$MMLC,ZMIN_$MMLC,
       ZFOCUS_$MMLC,ZTHICK_$MMLC,MIN_PLANE_$MMLC,MAX_PLANE_$MMLC;
$REAL LEAFNEG_$MMLC,LEAFPOS_$MMLC,INDEX_$MMLC,
      INDEX1_$MMLC,INDEX2_$MMLC,POS1_$MMLC,
      POS2_$MMLC,NEG1_$MMLC,NEG2_$MMLC,RNDM1_$MMLC;
$INTEGER NFIELDS_$MMLC,MODE_$MMLC,
      J_$MMLC,I_$MMLC,TOT_LEAF_$MMLC,
      IGNOREGAPS_$MMLC,MININD_$MMLC,MAXIND_$MMLC;
}

"Macro called upon each history to determine the field no. and set the"
"opening coordinates"
REPLACE {$SET_DYN_COMP} WITH {;
IF(MODE_$MMLC=1 | MODE_$MMLC=2)[
  $RANDOMSET RNDM1_$MMLC;
  DO I_$MMLC=1,NFIELDS_$MMLC [
       IF (INDEX_$MMLC(I_$MMLC)>=RNDM1_$MMLC)[
          INDEX2_$MMLC = INDEX_$MMLC(I_$MMLC);
          IF(MODE_$MMLC=1 & I_$MMLC>1)
                  INDEX1_$MMLC = INDEX_$MMLC(I_$MMLC-1);
          DO J_$MMLC=1,TOT_LEAF_$MMLC[
            NEG2_$MMLC = LEAFNEG_$MMLC(J_$MMLC+(I_$MMLC-1)*
                                       TOT_LEAF_$MMLC);
            POS2_$MMLC = LEAFPOS_$MMLC(J_$MMLC+(I_$MMLC-1)*
                                       TOT_LEAF_$MMLC);
            IF(MODE_$MMLC=1 & I_$MMLC>1)[
              NEG1_$MMLC = LEAFNEG_$MMLC(J_$MMLC+(I_$MMLC-2)*
                                       TOT_LEAF_$MMLC);
              POS1_$MMLC = LEAFPOS_$MMLC(J_$MMLC+(I_$MMLC-2)*
                                       TOT_LEAF_$MMLC);
              LEAFB_$MMLC(J_$MMLC)=NEG1_$MMLC+
                       (NEG2_$MMLC-NEG1_$MMLC)*
                       ((RNDM1_$MMLC-INDEX1_$MMLC)/
                        (INDEX2_$MMLC-INDEX1_$MMLC));
              LEAFA_$MMLC(J_$MMLC)=POS1_$MMLC+
                          (POS2_$MMLC-POS1_$MMLC)*
                          ((RNDM1_$MMLC-INDEX1_$MMLC)/
                         (INDEX2_$MMLC-INDEX1_$MMLC));
            ]
            ELSE[
              LEAFB_$MMLC(J_$MMLC)=NEG2_$MMLC;
              LEAFA_$MMLC(J_$MMLC)=POS2_$MMLC;
            ]
            IF(IGNOREGAPS_$MMLC=1 & IREJCT_GLOBAL>0)[
              IF(J_$MMLC=1)[
                MININD_$MMLC=J_$MMLC;
                MAXIND_$MMLC=J_$MMLC;
              ]
              ELSE[
                IF(LEAFB_$MMLC(J_$MMLC)<LEAFB_$MMLC(J_$MMLC-1))
                      MININD_$MMLC=J_$MMLC;
                IF(LEAFA_$MMLC(J_$MMLC)>LEAFA_$MMLC(J_$MMLC-1))
                      MAXIND_$MMLC=J_$MMLC;
              ]
            ]
          ]
"MIN_PLANE and MAX_PLANE determination changed"
          IF(IGNOREGAPS_$MMLC=1 & IREJCT_GLOBAL>0)[
```

```
              MIN_PLANE_$MMLC=LEAFB_$MMLC(MININD_$MMLC);
              MAX_PLANE_$MMLC=LEAFB_$MMLC(MAXIND_$MMLC);
          ]
          EXIT;
       ]
  ]
]

}


;
" $MMLC_FIND(REGION, DISTANCE)"
"======================================="
" this macro is used to determine the region #"
"{p1}: the return region value, local region"
"{p2}: the step dist which is used to calc. the particle final position"
REPLACE {$MMLC_FIND(#,#);} WITH {;
    XYFL(1)=XYL(1)+{P2}*UVL(1);
    XYFL(2)=XYL(2)+{P2}*UVL(2);
    ZFL=Z(NP)+{P2}*W(NP);
IF((W(NP)>0. & ZFL>=ZMAX_$MMLC) | (W(NP)<0. & ZFL<=ZFRONT_$MMLC)) [
    OUTOFCMFLAG=1;];
"put in = signs to get this to exit stepping loop in HOWFAR"
IF(N_GAP_$MMLC=1 & ZFL<=ZMIN_$MMLC & ZFL>=ZFRONT_$MMLC) [ {P1}=3; ]
ELSE ["inside leaf bank"
  LEAFIS=0;
  DO I = 1,TOT_LEAF_$MMLC [ "Determine which leaf we are in"
    TEMP1=MYREG_$MMLC(I,1)*(ZFL-ZTILT_$MMLC)+NYREG_$MMLC(I,1);
    TEMP2=MYREG_$MMLC(I,6)*(ZFL-ZTILT_$MMLC)+NYREG_$MMLC(I,6);
   IF(XYFL(1)>=TEMP1 & TEMP2>=XYFL(1)) [
     LEAFIS=I; EXIT; ];
   ];
  IF(LEAFIS=0) [{P1}=1;OUTOFMLCFLAG=1;] "assume going into air, set"
                                        "{P1}=2 for going into solid block"
  ELSE [
 "Z planes now are rotated"
    TEMP1=MZREG_$MMLC(LEAFIS,1)*XYFL(1)+NZREG_$MMLC(LEAFIS,1);
    TEMP2=MZREG_$MMLC(LEAFIS,7)*XYFL(1)+NZREG_$MMLC(LEAFIS,7);
    IF(ZFL<TEMP1)[
       NZ=1;
    ]
    ELSEIF(ZFL>TEMP2)[
       NZ=6;
    ]
    ELSE[
     DO I = 1,6 [ "Determine Z region"
      TEMP1=MZREG_$MMLC(LEAFIS,I)*XYFL(1)+NZREG_$MMLC(LEAFIS,I);
      TEMP2=MZREG_$MMLC(LEAFIS,I+1)*XYFL(1)+NZREG_$MMLC(LEAFIS,I+1);
      IF(ZFL>=TEMP1 & TEMP2>=ZFL) [
         NZ=I; EXIT; ];
     ];
    ]
    IF(NZ~=6 & W(NP)>0)[
     DO I=NZ+1,6[
       TEMP1=MZREG_$MMLC(LEAFIS,I)*XYFL(1)+NZREG_$MMLC(LEAFIS,I);
       IF(I=6 | TEMP1>ZFL) EXIT;
       NZ=I;
     ]
    ]
    TEMP1=MYREG_$MMLC(LEAFIS,1)*(ZFL-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,1);
    TEMP2=MYREG_$MMLC(LEAFIS,6)*(ZFL-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,6);
    IF(XYFL(1)<TEMP1)[
       NY=1;
    ]
    ELSEIF(XYFL(1)>TEMP2)[
       NY=5;
    ]
    ELSE[
       DO I =1,5 [ "Determine Y region"
         TEMP1=MYREG_$MMLC(LEAFIS,I)*(ZFL-ZTILT_$MMLC)+NYREG_$MMLC(LEAFIS,I);
         TEMP2=MYREG_$MMLC(LEAFIS,I+1)*(ZFL-ZTILT_$MMLC)+
                     NYREG_$MMLC(LEAFIS,I+1);
         IF(XYFL(1)>=TEMP1 & TEMP2>=XYFL(1)) [NY=I; EXIT;];
         ];
    ]
    IF(NY~=5 & XYFL(1)=MYREG_$MMLC(LEAFIS,NY+1)*(ZFL-ZTILT_$MMLC)+
               NYREG_$MMLC(LEAFIS,NY+1) & UVL(1)>0) [NY=NY+1;];
```

```
"Now determine X region index"
"Ortogonal projection of particle position to ZX boundaries"
"The second Z boundary corresponds to the upper boundary of the leaf"

IF(MOD(LEAFIS,2)~=0)[
ZXREG_$MMLC(1)=MZREG_$MMLC(LEAFIS,2)**2*XYFL(1)
                      +MZREG_$MMLC(LEAFIS,2)*XYFL(1)+NZREG_$MMLC(LEAFIS,2);
       ZXREG_$MMLC(1)=ZXREG_$MMLC(1)/(1+MZREG_$MMLC(LEAFIS,2)**2);
       ZXREG_$MMLC(2)=ZXREG_$MMLC(1)+HEDGEUP_U_$MMLC*
                      COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
       ZXREG_$MMLC(3)=ZXREG_$MMLC(2)+(ZTHICK_$MMLC-(HEDGEUP_U_$MMLC+
                      HEDGEDOWN_U_$MMLC))*COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
       ZXREG_$MMLC(4)=ZXREG_$MMLC(1)+ZTHICK_$MMLC*
                      COS(ATAN(MZREG_$MMLC(LEAFIS,2)));

       MAREG_$MMLC(1)=-WEDGEUP_U_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
       NAREG_$MMLC(1)=LEAFA_$MMLC(LEAFIS)+WEDGEUP_U_$MMLC-
                      ZXREG_$MMLC(1)*MAREG_$MMLC(1);
       MAREG_$MMLC(2)=WEDGEDOWN_U_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
       NAREG_$MMLC(2)=LEAFA_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MAREG_$MMLC(2);

       MBREG_$MMLC(1)=WEDGEUP_U_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
       NBREG_$MMLC(1)=LEAFB_$MMLC(LEAFIS)-WEDGEUP_U_$MMLC-
                      ZXREG_$MMLC(1)*MBREG_$MMLC(1);
       MBREG_$MMLC(2)=-WEDGEDOWN_U_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
       NBREG_$MMLC(2)=LEAFB_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MBREG_$MMLC(2);
]
ELSE[
    ZXREG_$MMLC(1)=MZREG_$MMLC(LEAFIS,2)**2*XYFL(1)
                      +MZREG_$MMLC(LEAFIS,2)*XYFL(1)+NZREG_$MMLC(LEAFIS,2);
       ZXREG_$MMLC(1)=ZXREG_$MMLC(1)/(1+MZREG_$MMLC(LEAFIS,2)**2);
       ZXREG_$MMLC(2)=ZXREG_$MMLC(1)+HEDGEUP_O_$MMLC*
                      COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
       ZXREG_$MMLC(3)=ZXREG_$MMLC(2)+(ZTHICK_$MMLC-(HEDGEUP_O_$MMLC+
                      HEDGEDOWN_O_$MMLC))*COS(ATAN(MZREG_$MMLC(LEAFIS,2)));
       ZXREG_$MMLC(4)=ZXREG_$MMLC(1)+ZTHICK_$MMLC*
                      COS(ATAN(MZREG_$MMLC(LEAFIS,2)));

       MAREG_$MMLC(1)=-WEDGEUP_O_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
       NAREG_$MMLC(1)=LEAFA_$MMLC(LEAFIS)+WEDGEUP_O_$MMLC-
                      ZXREG_$MMLC(1)*MAREG_$MMLC(1);
       MAREG_$MMLC(2)=WEDGEDOWN_O_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
       NAREG_$MMLC(2)=LEAFA_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MAREG_$MMLC(2);

       MBREG_$MMLC(1)=WEDGEUP_O_$MMLC/(ZXREG_$MMLC(2)-ZXREG_$MMLC(1));
       NBREG_$MMLC(1)=LEAFB_$MMLC(LEAFIS)-WEDGEUP_O_$MMLC-
                      ZXREG_$MMLC(1)*MBREG_$MMLC(1);
       MBREG_$MMLC(2)=-WEDGEDOWN_O_$MMLC/(ZXREG_$MMLC(4)-ZXREG_$MMLC(3));
       NBREG_$MMLC(2)=LEAFB_$MMLC(LEAFIS)-ZXREG_$MMLC(3)*MBREG_$MMLC(2);
];
"X region discrimination"

IF(ZFL<=ZXREG_$MMLC(2))[
       TEMP1=MBREG_$MMLC(1)*ZFL+NBREG_$MMLC(1);
       TEMP2=MAREG_$MMLC(1)*ZFL+NAREG_$MMLC(1);
       IF((XYFL(2)>TEMP1 & XYFL(2)<TEMP2) | (XYFL(2)=TEMP1 & UVL(2)>=0.0) |
       (XYFL(2)=TEMP2 & UVL(2)<0.0)) [NX=1;]
    ELSEIF((XYFL(2)<TEMP1) | (XYFL(2)=TEMP1 & UVL(2)<0.0)) [NX=2;]
    ELSEIF((XYFL(2)>TEMP2) | (XYFL(2)=TEMP2 & UVL(2)>=0.0)) [NX=3;]
]
ELSEIF(ZFL>ZXREG_$MMLC(2) & ZFL<=ZXREG_$MMLC(3))[
       TEMP1=LEAFB_$MMLC(LEAFIS);
       TEMP2=LEAFA_$MMLC(LEAFIS);
       IF((XYFL(2)>TEMP1 & XYFL(2)<TEMP2) | (XYFL(2)=TEMP1 & UVL(2)>=0.0) |
       (XYFL(2)=TEMP2 & UVL(2)<0.0)) [NX=1;]
    ELSEIF((XYFL(2)<TEMP1) | (XYFL(2)=TEMP1 & UVL(2)<0.0)) [NX=2;]
    ELSEIF((XYFL(2)>TEMP2) | (XYFL(2)=TEMP2 & UVL(2)>=0.0)) [NX=3;]
]
ELSEIF(ZFL>ZXREG_$MMLC(3))[
       TEMP1=MBREG_$MMLC(2)*ZFL+NBREG_$MMLC(2);
       TEMP2=MAREG_$MMLC(2)*ZFL+NAREG_$MMLC(2);
       IF((XYFL(2)>TEMP1 & XYFL(2)<TEMP2) | (XYFL(2)=TEMP1 & UVL(2)>=0.0) |
       (XYFL(2)=TEMP2 & UVL(2)<0.0)) [NX=1;]
    ELSEIF((XYFL(2)<TEMP1) | (XYFL(2)=TEMP1 & UVL(2)<0.0)) [NX=2;]
    ELSEIF((XYFL(2)>TEMP2) | (XYFL(2)=TEMP2 & UVL(2)>=0.0)) [NX=3;]
];
```

```
 {P1}=SUBINDEX_$MMLC(LEAFIS,NX,NY,NZ);
 ]; ]
;}
;

" $MMLC_MINDISTANCE;            "
"================================   "
" The following macro will give the nearest distance the particle"
" can travel along its given direction before it strikes the boundary"
" in the x, y or z direction."
"{P1}: The change value for ir region #"
" The distance equations come from the distance between the particle"
" position and the plane along the direction of movement IN 3-D!!!"

REPLACE {$MMLC_MINDISTANCE(#);} WITH {; "boundaries in Z direction"
IF(MOD(LEAFIS,2)~=0)["Unten"
       IF(NY=5 & (NZ>=2 & NZ<=5)) [ZLHS=2; ZRHS=6;]
    ELSE[ZLHS = NZ; ZRHS = NZ+1;]

]
ELSE["Oben"
       IF(NY=5 & (NZ>=2 & NZ<=5)) [ZLHS=2; ZRHS=6;]
       ELSE[ZLHS = NZ; ZRHS = NZ+1;]
]

"Z planes now are rotated"

IF(W(NP)-UVL(1)*MZREG_$MMLC(LEAFIS,ZRHS)~=0) [
    TEMP1=MZREG_$MMLC(LEAFIS,ZRHS)*XYFL(1)+NZREG_$MMLC(LEAFIS,ZRHS);
    ZP = (TEMP1-ZFL)/(W(NP)-UVL(1)*MZREG_$MMLC(LEAFIS,ZRHS));
]
ELSE [ ZP = 1.0E20; ];
IF(W(NP)-UVL(1)*MZREG_$MMLC(LEAFIS,ZLHS)~=0) [
       TEMP2=MZREG_$MMLC(LEAFIS,ZLHS)*XYFL(1)+NZREG_$MMLC(LEAFIS,ZLHS);
    ZN = (TEMP2-ZFL)/(W(NP)-UVL(1)*MZREG_$MMLC(LEAFIS,ZLHS));
]
ELSE [ ZN = 1.0E20; ];

IF( (ZP>=0.0) & (ZN>=0.0) ) [ ZDIST =MIN(ZP, ZN); ]
ELSE [ ZDIST = MAX(ZP, ZN); ]


IF(MOD(LEAFIS,2)~=0)["Unten"
  IF(NZ=3 & NY<=4) [LHS=1; J=LEAFIS;
    RHS=5; I=LEAFIS;
    ]
  ELSEIF(NZ=1) [LHS=1; J=LEAFIS;
    RHS=6; I=LEAFIS;
    ]
  ELSEIF(NZ=2 & (NY=2 | NY=3)) [LHS=2; J=LEAFIS;
    RHS=4; I=LEAFIS;
    ]
  ELSEIF(NZ=4 & (NY=1 | NY=2)) [LHS=1; J=LEAFIS;
    RHS=3; I=LEAFIS;
    ]
  ELSEIF(NZ=4 & (NY=3 | NY=4)) [LHS=3; J=LEAFIS;
    RHS=5; I=LEAFIS;
    ]
  ELSEIF(NZ=5 & NY<=4) [LHS=1; J=LEAFIS;
    RHS=5; I=LEAFIS;
    ]
  ELSEIF(NZ=6) [LHS=1; J=LEAFIS;
    RHS=6; I=LEAFIS;
    ]
  ELSE [LHS=NY; J=LEAFIS;
    RHS=NY+1; I=LEAFIS;
    ]
]
ELSE["Oben"
  IF(NZ=1) [LHS=1; J=LEAFIS;
    RHS=6; I=LEAFIS;
    ]
  ELSEIF(NZ=2 & NY<=4) [LHS=1; J=LEAFIS;
    RHS=5; I=LEAFIS;
    ]
  ELSEIF(NZ=3 & NY<=2) [LHS=1; J=LEAFIS;
    RHS=3; I=LEAFIS;
  ]
```

```
  ELSEIF(NZ=3 & (NY=3 | NY=4)) [LHS=3; J=LEAFIS;
    RHS=5; I=LEAFIS;
  ]
  ELSEIF(NZ=4 & NY<=4) [LHS=1; J=LEAFIS;
    RHS=5; I=LEAFIS;
  ]
  ELSEIF(NZ=5 & (NY=2 | NY=3)) [LHS=2; J=LEAFIS;
    RHS=4; I=LEAFIS;
  ]
  ELSEIF(NZ=6) [LHS=1; J=LEAFIS;
    RHS=6; I=LEAFIS;
  ]
  ELSE [LHS=NY; J=LEAFIS;
    RHS=NY+1; I=LEAFIS;
  ]
]

IF((UVL(1)-MYREG_$MMLC(I,RHS)*W(NP))~ = 0 ) [
       YP = (MYREG_$MMLC(I,RHS)*(ZFL-ZTILT_$MMLC)+NYREG_$MMLC(I,RHS)-XYFL(1));
    YP = YP/(UVL(1) - MYREG_$MMLC(I,RHS)*W(NP));
]
ELSE [ YP = 1.0E20; ];
IF((UVL(1) - MYREG_$MMLC(J,LHS)*W(NP))~ = 0 ) [
    YN = (MYREG_$MMLC(J,LHS)*(ZFL-ZTILT_$MMLC)+NYREG_$MMLC(J,LHS)-XYFL(1));
    YN = YN/(UVL(1) - MYREG_$MMLC(J,LHS)*W(NP));
]
ELSE [ YN = 1.0E20; ];

IF((YP>=0.0) & (YN>=0.0) ) [ YDIST =MIN(YP, YN); ]
ELSE [ YDIST = MAX(YP,YN);];

"Distance calculation in X"

IF(NX=1)[
       IF(UVL(2)-MBREG_$MMLC(1)*W(NP)~ = 0)[
              XN1=MBREG_$MMLC(1)*ZFL+NBREG_$MMLC(1)-XYFL(2);
              XN1=XN1/(UVL(2)-MBREG_$MMLC(1)*W(NP));
       ]
       ELSE [XN1=1.0E20;];
       IF(UVL(2)~ = 0)[
              XN2=LEAFB_$MMLC(LEAFIS)-XYFL(2);
              XN2=XN2/UVL(2);
       ]
       ELSE [XN2=1.0E20;];
       IF(UVL(2)-MBREG_$MMLC(2)*W(NP)~ = 0)[
              XN3=MBREG_$MMLC(2)*ZFL+NBREG_$MMLC(2)-XYFL(2);
              XN3=XN3/(UVL(2)-MBREG_$MMLC(2)*W(NP));
       ]
       ELSE [XN3=1.0E20;];

       IF(UVL(2)-MAREG_$MMLC(1)*W(NP)~ = 0)[
              XP1=MAREG_$MMLC(1)*ZFL+NAREG_$MMLC(1)-XYFL(2);
              XP1=XP1/(UVL(2)-MAREG_$MMLC(1)*W(NP));
       ]
       ELSE [XP1=1.0E20;];
       IF(UVL(2)~ = 0)[
              XP2=LEAFA_$MMLC(LEAFIS)-XYFL(2);
              XP2=XP2/UVL(2);
       ]
       ELSE [XP2=1.0E20;];
       IF(UVL(2)-MAREG_$MMLC(2)*W(NP)~ = 0)[
              XP3=MAREG_$MMLC(2)*ZFL+NAREG_$MMLC(2)-XYFL(2);
              XP3=XP3/(UVL(2)-MAREG_$MMLC(2)*W(NP));
       ]
       ELSE [XP3=1.0E20;];

       IF(XN1>=0.0)[
              XN=XN1;
              IF(XN2>=0.0)[
                     XN=MIN(XN,XN2);
              ]
              IF(XN3>=0.0)[
                     XN=MIN(XN,XN3);
              ]
       ]
       ELSEIF(XN2>=0.0 & XN3>=0.0)[
              XN=MIN(XN2,XN3);
       ]
```

```
        ELSE[
                XN=MAX(XN2,XN3);
        ]

        IF(XP1>=0.0)[
                XP=XP1;
                IF(XP2>=0.0)[
                        XP=MIN(XN,XP2);
                ]
                IF(XP3>=0.0)[
                        XP=MIN(XN,XP3);
                ]
        ]
        ELSEIF(XP2>=0.0 & XP3>=0.0)[
                XP=MIN(XP2,XP3);
        ]
        ELSE[
                XP=MAX(XP2,XP3);
        ]
]
ELSEIF(NX=2)[
        IF(UVL(2)~ = 0)[
                XN=-RMAX_CM(ICM_$MMLC)-XYFL(2);
                XN=XN/UVL(2);
        ]
        ELSE [XN=1.0E20;];

        IF(UVL(2)-MBREG_$MMLC(1)*W(NP)~ = 0)[
                XP1=MBREG_$MMLC(1)*ZFL+NBREG_$MMLC(1)-XYFL(2);
                XP1=XP1/(UVL(2)-MBREG_$MMLC(1)*W(NP));
        ]
        ELSE [XP1=1.0E20;];
        IF(UVL(2)~ = 0)[
                XP2=LEAFB_$MMLC(LEAFIS)-XYFL(2);
                XP2=XP2/UVL(2);
        ]
        ELSE [XP2=1.0E20;];
        IF(UVL(2)-MBREG_$MMLC(2)*W(NP)~ = 0)[
                XP3=MBREG_$MMLC(2)*ZFL+NBREG_$MMLC(2)-XYFL(2);
                XP3=XP3/(UVL(2)-MBREG_$MMLC(2)*W(NP));
        ]
        ELSE [XP3=1.0E20;];

        IF(XP1>=0.0)[
                XP=XP1;
                IF(XP2>=0.0)[
                        XP=MIN(XP,XP2);
                ]
                IF(XP3>=0.0)[
                        XP=MIN(XP,XP3);
                ]
        ]
        ELSEIF(XP2>=0.0 & XP3>=0.0)[
                XP=MIN(XP2,XP3);
        ]
        ELSE[
                XP=MAX(XP2,XP3);
        ]
]
ELSEIF(NX=3)[
        IF(UVL(2)-MAREG_$MMLC(1)*W(NP)~ = 0)[
                XN1=MAREG_$MMLC(1)*ZFL+NAREG_$MMLC(1)-XYFL(2);
                XN1=XN1/(UVL(2)-MAREG_$MMLC(1)*W(NP));
        ]
        ELSE [XN1=1.0E20;];
        IF(UVL(2)~ = 0)[
                XN2=LEAFA_$MMLC(LEAFIS)-XYFL(2);
                XN2=XN2/UVL(2);
        ]
        ELSE [XN2=1.0E20;];
        IF(UVL(2)-MAREG_$MMLC(2)*W(NP)~ = 0)[
                XN3=MAREG_$MMLC(2)*ZFL+NAREG_$MMLC(2)-XYFL(2);
                XN3=XN3/(UVL(2)-MAREG_$MMLC(2)*W(NP));
        ]
        ELSE [XN3=1.0E20;];

        IF(UVL(2)~ = 0)[
                XP=RMAX_CM(ICM_$MMLC)-XYFL(2);
```

```
                XP=XP/UVL(2);
        ]
        ELSE [XP=1.0E20;];

        IF(XN1>=0.0)[
                XN=XN1;
                IF(XN2>=0.0)[
                        XN=MIN(XN,XN2);
                ]
                IF(XN3>=0.0)[
                        XN=MIN(XN,XN3);
                ]
        ]
        ELSEIF(XN2>=0.0 & XN3>=0.0)[
                XN=MIN(XN2,XN3);
        ]
        ELSE[
                XN=MAX(XN2,XN3);
        ]
]

IF( (XP>=0.0) & (XN>=0.0) ) [ XDIST =MIN(XP, XN); ]
ELSE [ XDIST = MAX(XP,XN); ];

IF( ZDIST>=0.0) [TEMP=ZDIST;
    IF( YDIST>=0.0 ) [TEMP=MIN(TEMP,YDIST);];
    IF( XDIST>=0.0 ) [TEMP=MIN(TEMP,XDIST);];
  ]
ELSEIF ( (YDIST>=0.0) & (XDIST>=0.0) ) [TEMP=MIN(XDIST,YDIST);]
ELSE [TEMP=MAX(XDIST,YDIST);];
IF(TEMP<0) [OUTPUT NX,NY,NZ,LEAFIS; "should not happen"
     ('neg dist nx = ',I3,' NY= ',I3,' NZ= ',I3,' LEAF= ',I3);
   ];
{P1}=TEMP;
;} " End of $MMLC_MINDISTANCE macro"
;
" NEXT COMES HOWNEAR MACRO"
"========================="

REPLACE {$MMLC_CM_HOWNEAR(#);} WITH {
CALL HOWNEAR_$MMLC({P1});
}
;
"End of MMLC_macros.mortran"
```

REFERENCES

Attix, F. H. Introduction to Radiological Physics and Radiation Dosimetry. John Wiley and Sons, Inc., 1986.

Berger, M. J. Monte Carlo calculation of the penetration and diffusion of fast charged particles. Methods Comput. Phys., 1, 135-215, 1963.

Berger, M. J. and Wang, R. Multiple-scattering angular deflections and energy-loss straggling. Monte Carlo Transport of Electrons and Photons, Jenkins, T. M., Nelson, W. R., Rindi, A., Nahum, A. E., and Rogers, D. W. O. Eds., Plenum, New York, pp. 21-56, 1988.

Bielajew, A. F. Efficiency, statistics and sampling. National Research Council of Canada Report PIRS-0395, National Research Council of Canada, Ottawa, 1993

Bielajew A. F., Mohan R. and Chui C.S. Improved bremsstrahlung photon angular sampling in the EGS4 code system. National Research Council of Canada Report PIRS-0203 (1989)

Bielajew A. F. and Rogers D. W. O. Electron Monte Carlo simulation. National Research Council of Canada Report PIRS-0394, National Research Council of Canada, Ottawa, 1993.

Bjorken, J.D. and Drell, S.D. International Series in Pure and Applied Physics: Relativistic Quantum Fields. McGraw-Hill, New York, 1965

Boman, E. Radiotherapy Forward and Inverse Problem Appliying Boltzmann Transport Equation. PhD Thesis, University of Kuopio, Finland, 2007

Börgers C. The radiation therapy planning problem. *Comput. Radiol. Imag.* 110 1-16 1999

Bortfeld T, Oelke U and Nill S 2000 What is the optimum leaf width of a multileaf collimator? Med. Phys. 27 2494-502

Chilton, A. B. A note on the fluence concept, Health Phys. 34, 715-716, 1978.

Feller, W. An Introduction to Probability Theory and its Applications. Vol. 1, 3rd ed., Wiley, New York, 1967

Duderstadt, J. J. and Martin, W. R. Transport Theory, Wiley, New York, 1979.

Echner et al 2009. The design, physical properties and clinical utility of an iris collimator for robotic radiosurgery. Phys. Med. Biol. 54 (2009) 5359-5380.

Graves M N, Thompson A V, Martel M K, McShan D L, Fraass B A, 2001. Calibration and quality assurance for rounded leaf-end MLC systems. *Med Phys* **28** (11) 2227-2233.

Hartmann G H, Föhlisch F. Dosimetric characterization of a new miniature multileaf collimator. Phys. Med. Biol. 47 (2002) N171-N177.

Heath E, Seuntjens J. Development and validation of a BEAMnrc component module for accurate Monte Carlo modelling of the Varian dynamic Millennium multileaf collimator. *Phys. Med. Biol.* **48** (2003) 4045-4063.

ICRU (International Commission of Radiation Units and Measurements), Report 60: Fundamental Quantities and Units for Ionising Radiation, ICRU, Bethesda, MD, 1998.

James, F. RANLUX: A Fortran implementation of the high-quality pseudo-random number generator of Lüscher, *Computer Physics Communications*, 79 (1994) 111–114

Joint Committee for Guides in Metrology – Work Group 1. Evaluation of measurement data- Guide to the expression of uncertainty. JCGM 100:2008.

Kawrakow I and Bielajew A. F. On the condensed history technique for electron transport. Nuclear Instruments and Methods 142B, 253-280 (1998).

Kawrakow I and Rogers DWO. The EGSnrc Code System: Monte Carlo Simulation of Electron and Photon Transport. National Research Council of Canada, NRCC Report PIRS-701, 2003

Kawrakow I, Mainegra-Hing E and Rogers DWO. EGSnrcMP: the multi-platform environment for EGSnrc. National Research Council of Canada, NRCC Report PIRS-877, 2004

Kilby W et al 2009. Platforms for image-guided and adaptive radiation therapy. Image Guided and Adaptive Radiation Therapy. Ed R Timmerman and L Xing. Philadelphia, PA, Lippincott Williams and Wilkins) at press.

Koch H. W. and Motz J. W. Bremsstrahlung cross-section formulas and related data. Rev. Mod. Phys. 31, 920-55 1959

Lindeberg, J. W. Eine neue Herleitung des Exponentialgesetzes in der Wahrscheinlichkeitrechnung. Mathematische Zeitschrift, 15, 211-225, 1922

LoSasso T, Chui C-S, Ling C C, 1998. Physical and dosimetric aspects of a multileaf collimation system used in the dynamic mode for implementation intensity modulated radiotherapy. *Med. Phys.* **25** (10) 1919-1927.

Lüscher, M. A portable high-quality random number generator for lattice field theory calculations, *Computer Physics Communications*, 79 (1994) 100–110

Mayles P, Nahum A, RosenWald J. C. Handbook of Radiotherapy Physics: theory and practice. Taylor and Francis, Boca Raton FL, 2007.

McKie D and McKie C. Essentials of Crystallography. Blackwell Scientific Publications, Oxford 1992.

Møller, C. Zur Theorie des Durchgangs schneller Elektron durch Materie, Ann. Phys., 14, 531-585, 1932.

Pena J, González-Castaño D, Gómez F, Sánchez-Doblado F and Hartmann G H, 2007 Automatic determination of primary electron beam parameters in Monte Carlo Simulation. Med Phys 34 1076-1084.

Podgorsak, E. B. Radiation Oncology Physics: A Handbook for Teachers and Students. International Atomic Energy Agency, IAEA, Vienna, 2005.

Pönisch et al 2006. Properties of unflattened photon beams shaped by a multileaf collimator. Med. Phys. 33 (6).

Raeside, D. E. Monte Carlo principles and applications, Phys. Med. Biol., 21,181-197, 1976

Rogers DWO. Fifty years of Monte Carlo simulations for medical physics. Phys. Med. Biol. 51 (2006) R287–R301

Rogers DWO, Walters B, Kawrakow I. BEAMnrc Users Manual. National Research Council of Canada, NRCC Report PIRS-0509(A)revL, 2010

Sakurai, J. J. Advanced Quantum Mechanics. Addison-Wesley, Reading MA, 1967

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

In this approximately three years of doctoral studies many people have supported me and I would like to express them my gratitude.

I would like to thank my supervisor Prof. Günther Hartmann for accepting me in his research group. Thanks for all the advice, guidance and patience during this years. I specially appreciate your understanding and support during my staying in Chile, due to the birth of my son, and our nice conversations about family and parenting. I am really grateful and was an honor to work with you.

I thank Prof. Wolfgang Schlegel for his support as the head of our department at the DKFZ. Thanks for your help since the earlier days of the application process to the PhD and for agreeing to be the second Referee of my Thesis.

It is never an easy task to settle in a foreign country, especially with a family. I want to thank Simone Barthold-Beß, Marian and Franz-Joachim Kaiser for their kind support during our accommodation in Heidelberg and to help us to survive to the German burocracy throughout these years. Thanks to you we have been able to really enjoy our staying in this country.

Thanks to my friends and colleagues at the DKFZ, Martina Hub, Sonja Lahrmann, Andrés Vásquez, Ignacio Espinoza, Carine Legrand, Pamela Mena and Liu Hong for their friendly company during the different activities that we share during this time, in particular the lunches at the “Klinik” and the “Mensa” and the football evenings on spring and summer. I want to especially thank Ignacio for being a great friend and our “official interpreter” here in Germany and Carine for being a great office mate and for the proofreading of this thesis.

I want to specially acknowledge my family, in particular my parents Edgardo Dörner and María Loreto Yaksic. Muchas gracias por su apoyo incondicional durante toda mi vida y por creer siempre en mí. Ustedes siempre han sido un modelo a seguir para mí.

Finally, I have special words for my wife, María del Pilar. Pili, muchas gracias por tu amor y compañía durante todos estos años. Gracias por aceptar seguirme en esta aventura, a pesar de lo difícil que resultaría estar tan lejos de nuestros seres queridos. Todo este tiempo acá nos ha servido para conocernos mejor y crecer más como pareja. Gracias por guiarme y apoyarme en los momentos difíciles, por escucharme y por toda la paciencia que has tenido conmigo durante mi doctorado. Pero durante este tiempo no solo nos hemos desarrollado como pareja, también nuestra familia ha crecido. La llegada de Edgardito a nuestras vidas ha sido una experiencia maravillosa, y gracias a ustedes todo este tiempo ha sido mágico. Y ahora la llegada de María Elena nos traerá nuevos momentos de felicidad. No sé lo que nos depare el futuro, pero de lo que sí estoy seguro es que junto a ti ciertamente siempre será especial. Este trabajo te lo dedico especialmente a ti, mi amor.