Distribution de lumière dans un scintillateur

1. Introduction à la problématique des scintillateurs

Un scintillateur est un matériau ayant les propriétés d'émettre le rayonnement électromagnétique suite à l'absorption et l'interaction avec des particules énergétiques qui le traversent. Il existe une relation entre l'énergie déposée par les particules énergétiques interagissant avec le scintillateur et le nombre des particules produites par le scintillateur. Ainsi, cette relation permet de déduire l'énergie des particules interagissant avec le scintillateur et donc le type des particules.[1][2]

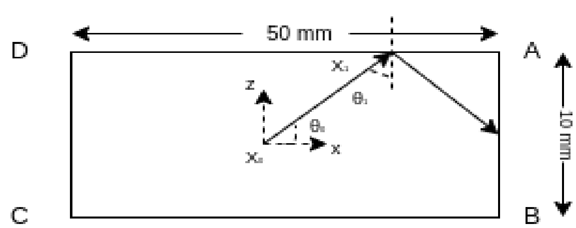
Il existe deux types des scintillateurs, les scintillateurs organiques et inorganiques, nous nous intéressons, dans la suite, a la modélisation d'un scintillateur inorganique solide. En particulier, on va simuler l'interaction des photons (émis par des rayons-γ) avec le scintillateur et donc leur trajectoire, dans le scintillateur, jusqu'à la sortie des photons de scintillateur ou leur absorption par des faces de scintillateur.

2. Objectifs de la simulation

L'objectif principal de la simulation de distribution de la lumière dans un scintillateur c'est de tracer le profil de lumière, c'est-à-dire de tracer la densité linéique de photons (en fonction de position) à la sortie du scintillateur.

La densité linéique de lumière devra être étudiée en fonction des différents revêtements possibles des faces latérales et de la face supérieure du scintillateur. Par exemple, les faces définies par les segments [AB] et [CD] pourront être soit totalement absorbantes le photon, soit réfléchissantes le photon avec un angle égale à l'angle d'incidence. Tandis que la face définie par le segment [DA] pourra être soit réfléchissante le photon avec un angle égale à l'angle d'incidence ou réfléchissante le photon avec un angle tiré aléatoirement selon la loi $\theta = arcsin(u)$ avec u étant la variable aléatoire tiré d'une distribution uniforme sur $[-1, 1[$.

La simulation devra être effectuée 10 milles fois, ensuite on pourra tracer la densité linéique de lumière, on s'entend d'avoir la distribution gaussienne des photons qui ont sorti.



(Figure 1: le scintillateur en 2D)

[1] Philippe Schwemling, Projet de physique numérique S1 Énoncés des projets, Distribution de lumière dans un scintillateur, l'Université de Paris

[2] https://fr.wikipedia.org/wiki/Scintillateur

3. **Modélisation de déplacement des photons dans le scintillateur**

Pour décrire le mouvement de photons dans un scintillateur on résout l'équation différentielle vectorielle, à vitesse constante,

$$\vec{v} = \frac{d\vec{x}}{dt}$$

par intégration on obtient l'équation décrivant le mouvement de photon entre deux instants,

$$\vec{x}_2 - \vec{x}_1 = \vec{v}(t_2 - t_1)$$

Qui peut aussi s'écrire comme:

$$\begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \end{pmatrix} = \begin{pmatrix} v_x(t_2 - t_1) \\ v_y(t_2 - t_1) \end{pmatrix} = \begin{pmatrix} v\cos(\theta)(t_2 - t_1) \\ vsin(\theta)(t_2 - t_1) \end{pmatrix}$$

Etant donné la position de photon à l'instant initial, l'angle de départ θ, la vitesse de photon et les dimensions de scintillateur on peut résoudre ce système d'équations et donc déduire les coordonnées de photons à la sortie de scintillateur.

Il est important de noter qu'un fois le photon est réfléchi par les faces de scintillateur, la composante orthogonale (à la face de réflection) du vecteur vitesse de photon change le signe. Par exemple, si le photon est réfléchi par la face définie par le segment $[AB]$ on effectue la transformation $(v_x, v_y) = (-v_x, v_y)$.

Algorithme

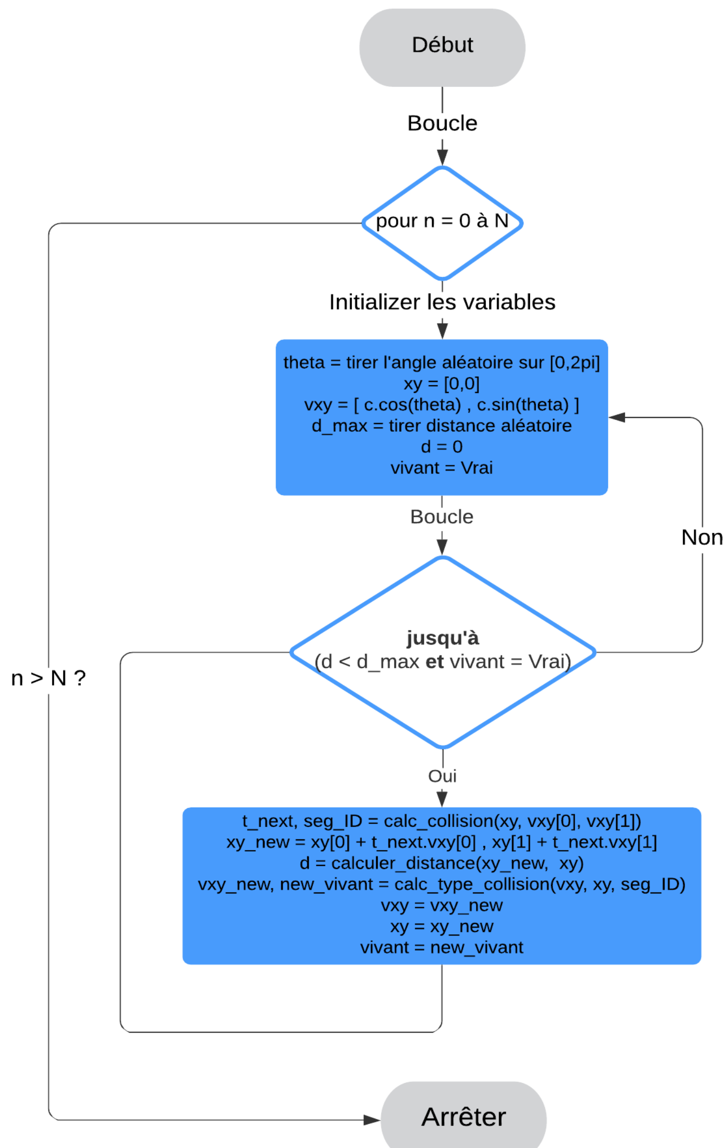
Il s'agit d'effectuer la simulation, en premier lieu en 2D *(Figure 1)*, ensuite en 3D. Dans la suite on décrit l'algorithme pour 2D, une fois la simulation est réalisée en 2D il faudra tenir quelques changements de la géométrie pour implémenter l'algorithme en 3D, néanmoins la logique restera inchangée.

La logique principale d'algorithme est représentée sur la *(Figure 2)*

1. On commence l'algorithme en tournant la boucle *pour n = 0 a N = 10 000*
    a. On initialize les variables principales:
        - *theta* - l'angle initial qui est tirée aléatoirement selon la loi uniforme sur [0, 2pi] (valeur)
        - *xy* - position (tableau)
        - *vxy* - vitesses initiales (tableau)
        - *d_max* - distance maximale de photon avant disparaitre (valeur) tirée selon la loi exponentielle d'espérance *l = 150mm*
        - *d* = 0 (valeur) distance totale parcourue par le photon
        - *vivant = Vrai* - (boolean) variable indiquant si le photon est absorbé

b. Pour le photon *n* donnée *boucler jusqu'à d < d_max et vivant = vrai*
   i. Appeler la fonction *calc_collision()* qui renvoie le plus court temps de collision - *t_next,* ainsi que le nom de parois de collision - *seg_ID*
   ii. Calculer les coordonnées de collision *xy_new*
   iii. Appeler la fonction *calculer_distance()* qui calcule la distance entre *xy* et *xy_new,* effectuer le valeur renvoyée à la variable *d*
   iv. Appeler la fonction *calc_type_collision()* qui renvoie les nouvelles composantes de la vitesse *vxy_new* et le boolean *new_vivant*

c. Vérifier si *vivant = vrai* ou *d > d_max,* si l'une des conditions est satisfaite aller vers *a.* (incrémenter n et recommencer la simulation pour le nouveau photon), sinon aller vers b (continuer la simulation jusqu'à photon sort et enregistrer ses coordonnées).



(Figure 2: l'algorithme de scintillateur)