



Développement d'un outil de simulation multi-échelle adapté au calcul des dommages radio-induits précoces dans des cellules exposées à des irradiations d'ions légers (proton et alpha)

Sylvain Meylan

► To cite this version:

Sylvain Meylan. Développement d'un outil de simulation multi-échelle adapté au calcul des dommages radio-induits précoces dans des cellules exposées à des irradiations d'ions légers (proton et alpha). Autre [cond-mat.other]. Université de Bordeaux, 2016. Français. <NNT : 2016BORD0184>. <tel-01420743>

HAL Id: tel-01420743

<https://tel.archives-ouvertes.fr/tel-01420743>

Submitted on 21 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

École doctorale SPI : Sciences Physiques et de l'Ingénieur

THÈSE DE DOCTORAT

pour obtenir le grade de docteur délivré par

l'Université de Bordeaux

Spécialité doctorale "Astrophysique, Plasmas, Noyaux"

présentée et soutenue publiquement par

Sylvain MEYLAN

le 21 octobre 2016

**Développement d'un outil de simulation multi-échelle
adapté au calcul des dommages radio-induits précoces
dans des cellules exposées à des irradiations d'ions légers
(proton et alpha)**

Directeur de thèse : **Sébastien INCERTI**

Co-encadrant de thèse : **Carmen VILLAGRASA**

Jury

Philippe Moretto,	Professeur	Président
Gérard Montarou,	Professeur	Rapporteur
Marie Davidkova,	Docteur	Rapporteur
Vincent Breton,	Directeur	Examineur
Jaime Angulo-Mora,	Directeur	Examineur

Recherches effectuées à l'IRSN/PRP-HOM/SDE/LDRI

Titre : Développement d'un outil de simulation multi-échelle adapté au calcul des dommages radio-induits précoces dans des cellules exposées à des irradiations d'ions légers (proton et alpha).

Résumé : Ce travail de thèse, réalisé dans le cadre des projets de recherche ROSIRIS (IRSN) et Geant4-DNA, porte sur la construction d'une simulation multi-échelle dédiée au calcul des dommages radio-induits précoces à l'ADN qui peuvent apparaître suite à l'irradiation d'un noyau cellulaire. L'outil développé s'appuie sur une version modifiée du code de Monte Carlo Geant4-DNA et est capable de simuler dans le détail le transport et les interactions physiques entre l'irradiation ionisante et la matière biologique (étape physique), la création d'espèces chimiques (étape physico-chimique) et les réactions et processus de diffusion de ces dernières (étape chimique). Durant la simulation de ces trois étapes, un modèle géométrique de l'ADN, décrivant l'ensemble du génome humain avec une précision moléculaire, est généré avec un nouveau logiciel développé dans le cadre de cette thèse : DnaFabric. Les premiers résultats obtenus pour des irradiations avec des protons et des ions alpha sont détaillés et comparés à des données de la littérature. Un bon accord est observé avec ces dernières illustrant ainsi la cohérence de l'ensemble de la simulation. L'influence très significative du critère de sélection utilisé pour identifier les dommages à l'ADN est également démontrée.

Mots clés : ADN, irradiation, génome humain, Geant4, Geant4-DNA, DnaFabric, SSB, DSB, cassure double brin, cassure simple brin, multi-échelle, 3D, simulation.

Title : Development of a multi-scale simulation tool for early radio-induced damage assessment in cells exposed to light ions irradiations (proton and alpha).

Abstract : This work was performed in the frame of the ROSIRIS (IRSN) and Geant4-DNA research projects and describes the development of a simulation tool to compute radio-induced early DNA damages in a cell nucleus. The modeling tool is based on a modified version of the Monte Carlo code Geant4-DNA and is able to simulate the physical interactions between ionizing particles and the biological target (physical stage), the creation of chemical species within the cell nucleus (physico-chemical stage) as well as the reactions and diffusion processes of these chemical species (chemical stage). During all the simulation, a geometrical model that describes the DNA content of a human diploid cell nucleus is taken into account. This model was generated with a new software (DnaFabric) developed in the frame of this work and has a molecular level of detail. The first results (in term of DNA strand breaks) obtained with this tool are detailed and compared with experimental data from the literature. The good agreement between the simulation results and those data shows the coherence of our modeling. The significant influence of the selection criteria used to identify the DNA damages is also demonstrated.

Keywords : DNA, irradiation, human genome, Geant4, Geant4-DNA, DnaFabric, SSB, DSB, double strand break, single strand break, multi-scale, 3D, simulation.

Unité de recherche

Institut de Radioprotection et de Sécurité Nucléaire (IRSN)

Laboratoire LDRI (PRP-HOM/SDE/LDRI)

31, avenue de la Division Leclerc

92260 Fontenay aux Roses

Remerciements

La fin de ces travaux de thèse clôture trois années de ma vie durant lesquelles j'ai eu la chance de travailler dans un environnement stimulant, dynamique mais également bienveillant et chaleureux. Pour cela, je tiens à remercier l'ensemble des personnes avec lesquelles j'ai discuté, questionné, appris, douté et persévéré.

Je salue tout particulièrement :

L'encadrement

J'ai eu la chance de bénéficier d'un excellent encadrement durant ces trois années de thèse. La disponibilité et la bienveillance de Carmen Villagrasa, mon encadrante au quotidien, et de Sébastien Incerti, mon directeur de thèse, ont été des facteurs essentiels à la réussite de ces travaux.

Le jury

Je tiens à adresser mes remerciements aux membres du jury pour leur relecture méticuleuse de ce manuscrit ainsi que leurs conseils et questions avisées. Je salue Gérard Montarou et Marie Davidkova pour avoir accepté de rapporter de manuscrit, Philippe Moretto pour l'avoir présidé et, enfin, Vincent Breton et Jaime Angulo-Mora pour avoir examiné attentivement ce travail.

Les collaborations

Ce travail s'est nourri et construit à partir de réflexions menées collectivement et des efforts réalisés par le passé dans le cadre de plusieurs collaborations et projets de recherche, je pense notamment à Geant4, Geant4-DNA, ROSIRIS et BioQuaRT. Pouvoir évoluer au sein de projets aussi solides tant scientifiquement qu'humainement fut un honneur.

Les collègues

Qu'ils soient, stagiaires, doctorants, chercheurs, ingénieurs ou support, c'est grâce à la sympathie et à l'amitié de mes collègues que ces trois années se sont déroulées dans la bonne humeur. J'ai une pensée émue pour ceux d'entre eux qui ont partagé mon bureau au cours de ces trois années.

Le cercle personnel

Ma famille et ma compagne m'ont soutenu, écouté et conseillé dans les moments de doute et de remise en question. A ce titre, ils méritent amplement ma gratitude et leur place au sein de ces remerciements.

Table des matières

Liste des figures	v
Liste des tableaux	vi
Liste des algorithmes	vii
Introduction	1
Références	4
1 Calcul des dommages radio-induits à l'ADN	5
1.1 Effets biologiques des rayonnements ionisants	7
1.1.1 Caractérisation physique du dépôt d'énergie	7
1.1.2 Impact des rayonnements ionisants au niveau cellulaire	9
1.1.3 Effets à l'échelle de l'individu	11
1.1.4 Objectif de ce travail de thèse	13
1.2 Transport de particules avec la méthode de Monte Carlo	14
1.2.1 Méthode de Monte Carlo appliquée au transport de particules	14
1.2.2 Le code de simulation Geant4	17
1.2.3 L'extension Geant4-DNA	23
1.2.4 Utiliser Geant4/Geant4-DNA	26
1.3 Simulation de la radiolyse de l'eau	29
1.3.1 Généralités	29
1.3.2 La chimie dans Geant4-DNA	30
1.3.3 Utiliser le module chimie de Geant4-DNA	37
1.4 L'ADN et les dommages radio-induits	38
1.4.1 Structure de l'ADN : du nucléotide à la fibre de chromatine	39
1.4.2 Distribution spatiale de la chromatine dans le noyau cellulaire	40
1.4.3 Les dommages à l'ADN	43
1.4.4 Modélisations de l'ADN pour le calcul des dommages radio-induits	46
Références	54
2 Modèle géométrique de l'ADN	55
2.1 Développement de DnaFabric	57
2.1.1 Introduction	57
2.1.2 Architecture du projet	58
2.1.3 Implémentation d'un modèle géométrique dans DnaFabric	61
2.1.4 Visualisation 3D	66
2.2 Modélisation de l'ADN dans DnaFabric	73
2.2.1 De la molécule au nucléosome	73
2.2.2 Structures de haut niveau formées par l'ADN	78
2.3 Modélisation de noyaux cellulaires	81
2.3.1 Modélisation du génome humain	82

2.3.2	Algorithmes de remplissage du noyau	86
2.3.3	Les noyaux cellulaires	89
	Références	95
3	Simulation de l'irradiation d'un noyau cellulaire	97
3.1	Simulation de l'étape physique	99
3.1.1	Prise en compte de l'ADN lors du transport des particules	99
3.1.2	Choix d'une méthode pour l'attribution d'une interaction à un composant de l'ADN	107
3.1.3	Introduction du noyau cellulaire dans la simulation	110
3.1.4	Génération de données lors de la simulation	113
3.2	Simulation des étapes physico-chimique et chimique	116
3.2.1	Prise en compte du modèle géométrique	116
3.2.2	Génération de données lors de la simulation	120
3.2.3	Quelques paramètres spécifiques à l'étape chimique	121
3.3	Organisation de la simulation des trois étapes	122
3.3.1	Contenu et architecture	122
3.3.2	Utilisation de "FullSim"	124
	Références	130
4	Dommmages à l'ADN obtenus par la simulation de l'irradiation d'un noyau cellulaire	131
4.1	Irradiation d'un noyau de cellule fibroblaste : comparaison avec des données de la littérature	133
4.1.1	Configuration des simulations	134
4.1.2	Résultats : comparaison des résultats de la simulation avec les données publiées sur les fragments d'ADN	137
4.1.3	Discussion	143
4.2	Influence des sections efficaces de l'ADN	146
4.2.1	Configuration des simulations	146
4.2.2	Résultats : influence du matériau composite ADN sur le nombre d'ionisations à l'échelle de 10 paires de nucléotides	148
4.2.3	Résultats : influence du matériau composite ADN à l'échelle du noyau de cellule fibroblaste	149
4.2.4	Discussion	152
4.3	Sensibilité de la simulation	155
4.3.1	Configuration des simulations	155
4.3.2	Résultats : sélection des dommages directs	158
4.3.3	Résultats : durée simulée de l'étape chimique	159
4.3.4	Résultats : influence des voxels isolés	160
4.3.5	Discussion	162
	Références	168
	Conclusion	169
	Références	174
A	Sections efficaces de l'ADN pour les électrons	I
B	Sections efficaces de l'ADN pour les protons	VII
C	Distributions de la complexité des DSB simulés avec des protons primaires	XI

Liste des figures

1.1	Variation du rapport $\frac{\epsilon}{m}$ en fonction de la masse m du site étudié	8
1.2	Les dépôts d'énergie de protons et alphas obtenus avec Geant4-DNA	10
1.3	La survie cellulaire en fonction de la dose absorbée obtenue suite à un même type de cellules avec des doses différentes et deux rayonnements distincts	11
1.4	Les effets radio-induits repartis en cinq grandes étapes	14
1.5	Représentations de sections efficaces atomiques dans un plan	15
1.6	Liste de codes de Monte Carlo	18
1.7	Articulation des concepts de run, event, track et step lors d'une simulation Geant4	21
1.8	Décomposition d'un step	21
1.9	Les principaux modules de Geant4	22
1.10	Logigramme simplifié des méthodes utilisées lors d'une simulation Geant4	28
1.11	Schéma représentant la création d'un électron solvaté.	32
1.12	Deux paires de nucléotides : A-T et G-C	39
1.13	La double hélice de l'ADN	39
1.14	Vues d'un nucléosome avec la double hélice de l'ADN enroulée autour du complexe de protéines histones	40
1.15	Organisation des nucléosomes au sein d'une fibre de chromatine selon les modèles "solénoïde" et "zizag"	40
1.16	Les différentes structures de la chromatine lors du cycle cellulaire	41
1.17	Les différents niveaux de compaction de la chromatine	42
1.18	Cassures simple brin de l'ADN	44
1.19	Cassures double brin de l'ADN	44
1.20	Les cassures double brin de l'ADN détectées avec des foci	46
1.21	Deux représentations de l'ADN avec des niveaux de détails différents	47
1.22	Modèle géométrique de l'ADN contenu dans un noyau de cellule fibroblaste	47
2.1	Logigramme simplifié de DnaFabric	60
2.2	13 pavés le long de deux segments	61
2.3	Arbre et graphe	64
2.4	Organisation hiérarchique et empreintes mémoires	65
2.5	Triangle et vertices	67
2.6	Primitive géométrique	68
2.7	Pipeline graphique	69
2.8	Primitive géométrique et quatre niveaux de détail (LOD)	69
2.9	Logigramme du moteur de rendu	72
2.10	Fonctionnement du transfert de données CPU/GPU	73
2.11	Structure hiérarchique du modèle ADN développé dans DnaFabric	74
2.12	Construction d'une paire de nucléotides	75

2.13	Différents types de double hélice formés par l'ADN	75
2.14	Double hélice droite de type B-DNA construite dans DnaFabric.	76
2.15	Nucléosome construit dans DnaFabric	77
2.16	Deux nucléosomes reliés par un linker	77
2.17	Fibres d'hétérochromatine et d'euchromatine	78
2.18	Les 5 voxels de DnaFabric	80
2.19	Un filament d'ADN	81
2.20	Organisation hiérarchique de l'ADN dans un noyau cellulaire	82
2.21	Un génome humain de type mâle, dans l'état condensé et contenu dans un noyau sphérique	85
2.22	Un génome humain de type mâle obtenu après une simulation de relaxation	85
2.23	Construction d'une boucle de chromatine dans un domaine sphérique	88
2.24	Algorithme de remplissage pour maximiser la densité d'ADN	89
2.25	Représentation 3D d'un noyau cellulaire au travers de trois niveaux de zoom	90
2.26	Modélisation d'un noyau de cellule fibroblaste	91
2.27	Modélisation d'un noyau de cellule lymphocyte	92
2.28	Modélisation d'un noyau de cellule endothéliale	92
3.1	Distributions de probabilité du nombre d'ionisations	105
3.2	M1 et F2 calculés avec Geant4 et PTra	107
3.3	10 paires de nucléotides représentant la configuration géométrique utilisée dans le premier ensemble de simulations	108
3.4	Nombre moyen d'ionisations enregistrées dans 10 paires de nucléotides en fonction de l'énergie et du mode d'identification.	109
3.5	Illustration du processus permettant d'éliminer les chevauchements	111
3.6	Identification d'un volume avec son nom et l'identifiant de la paire de nucléotide	114
3.7	Organisation des données issues de phys_geo pour obtenir la liste des simulations intermédiaires à exécuter	117
3.8	Les différents modules contenus dans la chaîne de calcul FullSim	124
3.9	Logigramme du fonctionnement de la chaîne de calcul FullSim	127
4.1	Dimensions du noyau de cellule fibroblaste et configuration de la source	134
4.2	Regression linéaire effectuée à partir de plusieurs irradiations avec des protons de 5 MeV	136
4.3	Nombres moyens de DSB simulés et détectés expérimentalement	138
4.4	Nombre moyen de DSB simulé avec différents codes de structure de trace issu de la littérature et de FullSim.	139
4.5	Distributions normalisées du nombre de DSB calculés par proton primaire (event) avec une taille minimale de fragment de 10000 bp	140
4.6	Nombre de DSB en fonction de l'énergie du proton incident et d'une taille minimale de fragment	141
4.7	Nombre de SSB calculés en simulant l'irradiation d'un noyau de cellule fibroblaste avec des protons	143
4.9	Nombre moyen d'ionisations enregistrées dans 10 paires de nucléotides en fonction de l'énergie et du matériau pris en compte	149
4.10	Nombre moyen d'ionisations enregistrées dans 10 paires de nucléotides ($\rho = g/cm^3$) en fonction de l'énergie et du matériau utilisé pour remplir l'ADN	149

4.11	Nombre moyen de DSB/Gy/Gbp générés par des protons de différentes énergies dans un noyau de cellule fibroblaste contenant de l'ADN. L'ADN est constitué d'eau liquide puis du matériau composite ADN	150
4.12	Nombre moyen de SSB en fonction du TEL calculés avec FullSim dans un noyau de cellule fibroblaste : l'ADN est constitué d'eau liquide puis du matériau composite représentatif	151
4.13	Le nombre de DSB/Gy/Gbp en fonction de la densité du matériau composite ADN et pour une irradiation du noyau de fibroblaste avec des protons primaires de 1 MeV	152
4.14	Deux fibres de chromatine construites avec DnaFabric : une fibre d'hétérochromatine et une fibre constituée de 4 voxels "avant" consécutifs.	157
4.15	Influence de la présence d'un critère de sélection des SSB _{direct} sur le nombre moyen de DSB/Gy/Gbp calculé	158
4.16	Effet sur le nombre moyen de DSB/Gy/Gbp calculé à partir de deux critères de sélection concernant les SSB _{direct}	159
4.17	Application de 3 critères de sélection différents pour l'identification des SSB _{direct} lors du calcul du nombre moyen de DSB/Gy/Gbp	160
4.18	Le nombre moyen de DSB/event calculés avec FullSim en fonction de différentes durées de l'étape chimique	161
4.19	Le nombre de DSB calculé par particule primaire avec FullSim dans deux géométries distinctes : une suite de 4 voxels "avant" et une fibre continue de 90 nucléosomes	162
A.1	Sections efficaces totales d'ionisation (électrons) pour les matériaux précurseurs de l'ADN	I
A.2	Sections efficaces totales d'ionisation (électrons) pour les matériaux ADN	II
A.3	Sections efficaces totales d'excitation (électrons) pour les matériaux précurseurs de l'ADN	II
A.4	Sections efficaces totales d'excitation (électrons) pour les matériaux ADN	III
A.5	Sections efficaces totales de diffusion élastique (électrons) pour les matériaux précurseurs de l'ADN	III
A.6	Sections efficaces totales de diffusion élastique (électrons) pour les matériaux ADN	IV
A.7	Sections efficaces différentielles d'ionisation (électrons) pour les matériaux précurseurs de l'ADN	V
A.8	Sections efficaces différentielles de diffusion élastique (électrons) pour les matériaux précurseurs de l'ADN	VI
B.1	Sections efficaces totales d'ionisation (proton) pour les matériaux précurseurs de l'ADN	VII
B.2	Sections efficaces totales d'ionisation (proton) pour les matériaux ADN	VIII
B.3	Sections efficaces différentielles d'ionisation (protons) pour les matériaux précurseurs de l'ADN	IX
C.1	Distribution de la complexité des DSB simulés avec FullSim dans un noyau de cellule fibroblaste pour différentes énergies de protons primaires : 20, 10, 5 et 4 MeV	XI
C.2	Distribution de la complexité des DSB simulés avec FullSim dans un noyau de cellule fibroblaste pour différentes énergies de protons primaires : 3 ; 2 ; 1,5 ; 1 ; 0,7 et 0,5 MeV	XII

Liste des tableaux

1.1	Syndrome d'irradiation aiguë	12
1.2	Modèles physiques de Geant4-DNA et leurs domaines de validité en énergie	25
1.3	Les canaux de dissociation présents dans Geant4-DNA pour une molécule d'eau liquide excitée	31
1.4	Les coefficients de diffusion des espèces chimiques pris en compte dans Geant4-DNA	33
1.5	Les coefficients de réaction pris en compte dans Geant4-DNA	36
1.6	La configuration des pas en temps dans Geant4-DNA	36
2.1	Balises utilisées dans le fichier ".fab2g4dna"	66
2.2	Tableau des primitives à afficher	71
2.3	Volumes des sphères représentant les molécules des paires de nucléotides	74
2.4	Caractéristiques des fibres d'hétérochromatine et d'euchromatine	79
2.5	Caractéristiques des 5 voxels implémentés dans DnaFabric	80
2.6	Les 23 paires de chromosomes du génome humain	83
2.7	Caractéristiques des noyaux générés avec DnaFabric	91
3.1	Les facteurs appliqués aux sections efficaces des précurseurs pour obtenir celles des constituants de l'ADN	101
3.2	Informations concernant les couches électroniques des constituants de l'ADN et de leurs précurseurs	101
3.3	Matériau composite représentatif de l'ADN	106
3.4	Les réactions implémentées dans chem_geo entre l'ADN et les espèces chimiques	119
4.1	Énergies cinétiques et TEL des protons utilisés pour irradier le noyau cellulaire	135
4.2	Nombre moyen de DSB/Gy/Gbp simulés en fonction du matériau utilisé pour l'ADN (eau liquide ou matériau composite ADN) et du critère de sélection des SSB _{direct} choisi	152
4.3	Nombre de paires de nucléotides totalet par unité de longueur pour une fibre d'hétérochromatine et une fibre constituée de 4 voxels "avant" consécutifs	157
4.4	Augmentation du nombre de DSB moyen calculé avec FullSim lorsque la durée de l'étape chimique varie de 2,5 ns au seuil de stabilisation du nombre moyen de DSB	162

Liste des algorithmes

1.1	Version simplifiée du transport de particule avec la méthode de Monte Carlo	19
2.1	Construction d'une géométrie avec l'approche "équation"	63
2.2	Construction d'une géométrie avec l'approche "objet"	64
2.3	Relier deux cases d'un réseau cubique avec un filament d'ADN.	87
3.1	Algorithme de découpage des sphères Geant4 pour éviter tout chevauchement	112

Introduction

La compréhension du lien entre la caractérisation physique d'un rayonnement ionisant et son effet biologique constitue un domaine de recherche actif et particulièrement transversal. Ce domaine de recherche est apparu suite à l'utilisation des rayonnements ionisants dans le cadre de diverses activités : industrie du nucléaire, médecine, applications militaires et, également, exploration spatiale. Les tentatives visant à établir un tel lien se sont appuyées sur les grandeurs physiques communément utilisées pour caractériser les rayonnements ionisants, soit principalement la dose absorbée et le kerma. L'idée était de corréliser ces grandeurs physiques aux effets observés lorsque des populations cellulaires étaient irradiées lors d'expériences de radiobiologie [BOLUS, 2001]. Ces expériences ont conduit au concept de RBE ("Relative Biological Effectiveness") qui rend compte de l'effet biologique d'un type de rayonnement relativement à celui induit par un rayonnement de référence. Les différents RBE obtenus expérimentalement ont mis en évidence le fait que pour un système biologique donné et une dose absorbée fixe alors les effets des rayonnements ionisants dépendent à la fois de l'énergie et du type de rayonnement utilisé (ions carbone, ions alpha, photons etc.). Cette dépendance a une faible influence lorsque les doses reçues sont importantes (i.e. de l'ordre de grandeur du Gy) et ceci explique pourquoi la dose absorbée est toujours la grandeur de référence utilisée, par exemple, dans le monde médical. Toutefois, cette dépendance prend de l'importance lorsque les doses diminuent et que les effets biologiques deviennent moins évidents à observer. Une telle situation apparaît souvent dans le cadre de la radioprotection mais également, par exemple, lors des radiothérapies où les tissus sains situés autour des tumeurs à irradier reçoivent de faibles doses. L'apport potentiel d'une compréhension fine des processus à l'origine des effets biologiques prend alors tout son sens : des modèles prédictifs plus précis pourraient être établis et améliorer ainsi la gestion des risques encourus lors des traitements par radiothérapie, de l'exploration spatiale ou encore lors de catastrophes de nature nucléaire. Dans ce cadre, les grandeurs physiques usuelles ne sont pas adaptées et cela se traduit par une dépendance du RBE au type de rayonnement utilisé et à son énergie [KRÄMER, 2009]. Cette dépendance provient des limitations de la dose absorbée qui est la grandeur physique sur laquelle le RBE est basé. En effet, la dose absorbée ne prend pas en compte la répartition spatiale des dépôts d'énergie induits dans la matière par un rayonnement ionisant. Et cette répartition est spécifiquement liée aux effets biologiques en raison de la taille nanométrique des entités biologiques (cibles) contenues dans une cellule et sensible aux rayonnements ionisants. Par exemple, l'ADN est caractérisé par des dimensions de l'ordre du nanomètre et s'avère particulièrement sensible aux dommages induits par les rayonnements ionisants. L'échelle nanométrique des cibles considérées implique d'étudier la répartition des dépôts d'énergie à une échelle similaire. Hors, la répartition des dépôts d'énergie est, à cette échelle, stochastique par nature et c'est pourquoi son étude est hors de portée des grandeurs physiques précédemment mentionnées. En effet, ces grandeurs physiques sont des moyennes construites à partir de l'ensemble de dépôts d'énergie individuels et, de part ce fait, elles ne permettent pas d'accéder à des informations détaillées concernant les dépôts d'énergie individuels. La microdosimétrie [ROSSI, 1959] est un domaine de recherche basé sur l'étude de grandeurs physiques spécifiquement conçues pour fournir des informations sur les aspects

stochastiques de la répartition des dépôts d'énergie. Toutefois, ces informations sont tout de même intégrées car la microdosimétrie étudie cela à l'échelle du micromètre qui est plusieurs ordres de grandeurs au delà du nanomètre. Finalement, c'est la nanodosimétrie qui permet d'accéder à l'ensemble des informations associées aux dépôts d'énergie et à leur répartition spatiale détaillée. Notons que des dispositifs expérimentaux appelés nano-dosimètres sont utilisés depuis quelques années pour la recherche. Ces derniers permettent de mesurer le nombre d'ionisations ayant lieu dans un volume de taille micrométrique par particule incidente. Ajoutons également que les répartitions spatiales de ces dépôts peuvent être simulées au moyen de codes de simulation Monte Carlo tels que, par exemple, PARTRAC [FRIEDLAND et al., 2011], KURBUC [NIKJOO et al., 2006], PTra [GROSSWENDT, 2002] et Geant4-DNA [BERNAL et al., 2015; INCERTI et al., 2010]. Dans ce contexte, l'IRSN (Institut de Radioprotection et de Sécurité Nucléaire) a débuté un projet de recherche nommé ROSIRIS (Radiobiologie des Systèmes Intégrés pour l'optimisation des traitements utilisant des rayonnements ionisants et évaluation du RISque associé) dont le but est d'améliorer la connaissance sur les mécanismes d'initiation et de progression des complications des radiothérapies pour, ainsi, améliorer les modèles d'évaluation des risques associés. Il s'agit d'un projet transdisciplinaire impliquant des physiciens, biologistes et chimistes dont la stratégie globale consiste à tenter d'établir un lien entre des événements précoces radio-induits, issus de modélisations biophysiques, et leurs conséquences tardives au niveau cellulaire et tissulaire. L'établissement de ce lien vise à acquérir de nouvelles capacités de prédictions concernant l'apparition de complications sur les organes à risques. Sur le long terme, le projet ROSIRIS devrait permettre d'améliorer les connaissances concernant les effets secondaires des rayonnements ionisants, d'optimiser et personnaliser les protocoles de radiothérapie et d'envisager de nouvelles pistes de traitement. Le projet ROSIRIS est divisé en trois axes de recherche (source : www.irsn.fr) :

- Axe 1 Simuler l'irradiation de cellules de manière à établir des corrélations entre irradiation et dommages biologiques précoces.
- Axe 2 Démontrer à l'échelle d'un tissu, l'existence d'un lien entre les effets biologiques tardifs d'une irradiation et identifier les molécules impliquées.
- Axe 3 Valider les hypothèses des mécanismes à l'origine du lien entre les effets précoces et les effets tardifs in-vivo sur des souris transgéniques.

Le travail de thèse présenté dans ce manuscrit s'inscrit spécifiquement dans le premier axe du projet, à savoir les modélisations biophysiques destinées à simuler des événements précoces radio-induits pour les corrélés à des effets biologiques. Plus précisément, l'objectif de ce travail de thèse est de simuler les dommages radio-induits précoces à l'ADN à l'échelle d'un noyau cellulaire contenant un génome humain, soit environ 6,4 milliards de paires de nucléotides. Une telle simulation requiert la prise en charge de trois étapes temporellement distinctes dans la description de l'interaction rayonnement-ADN : les étapes physique, physico-chimique et chimique. Lors de l'étape physique, les particules constituant le rayonnement ionisant traversent la matière et, à cette occasion, peuvent interagir spécifiquement avec l'ADN et l'endommager. De tels dommages à l'ADN sont qualifiés de "directs". Les interactions des particules avec l'eau liquide contenue dans un noyau cellulaire ont une certaine probabilité d'ioniser ou d'exciter les molécules d'eau qui se retrouvent alors dans des états instables. Ces états instables peuvent alors provoquer la dissociation de ces molécules d'eau en plusieurs espèces chimiques. Ce processus de dissociation a lieu lors de l'étape physico-chimique. Les espèces chimiques ainsi créées vont ensuite diffuser dans le noyau cellulaire et réagir entre elles et, parfois, avec l'ADN. Il s'agit alors de l'étape chimique. Parmi toutes les réactions prise en compte

lors de cette étape, certaines impliquent l'ADN et peuvent causer des dommages qualifiés cette fois d'"indirects".

Dans ce contexte, le **premier chapitre** de cette thèse constitue un état de l'art du calcul des dommages radio-induits à l'ADN par simulation de Monte Carlo. Dans un premier temps, la compréhension actuelle des effets des rayonnements ionisants est introduite puis le transport des particules avec la méthode de Monte Carlo est explicité. Ensuite, la radiolyse de l'eau est présentée sa simulation étant basée sur des développement récemment effectués au sein du code de simulation Geant4 (Geant4-DNA). Enfin, la structure de la chromatine contenue dans un noyau cellulaire est décrite avec les différents types de dommages que peut subir l'ADN lors d'une irradiation.

La modélisation géométrique de l'ADN dans le cadre d'une telle simulation est tout particulièrement centrale car ce dernier est considéré comme la cible privilégiée de l'irradiation simulée. Le **second chapitre** de ce travail de thèse est donc entièrement dédié à cette modélisation. Une méthode permettant de construire des modélisations géométriques de l'ADN complexes et flexibles est introduite dans ce chapitre. Cette méthode est inspirée de technologies développées en informatique graphique et est implémentée dans un outil spécifiquement développé dans le cadre de ce travail. Les modélisations géométriques de l'ADN ainsi construites peuvent représenter les 6,4 milliards de paires de nucléotides contenues dans un noyau de cellule humaine (diploïde).

Le **troisième chapitre** décrit la chaîne de calcul basée sur Geant4 (Geant4-DNA) et développée dans ce travail pour simuler les étapes physique, physico-chimique et chimique déclenchées lors de l'irradiation d'un noyau cellulaire. Cette simulation intègre la modélisation géométrique détaillée de l'ADN introduite dans le second chapitre et est donc capable de simuler les dommages précoces occasionnés à l'ADN. Une des spécificités de cette chaîne de calcul est la possibilité de prendre en compte l'ADN en tant que matériau lors du transport de particule (méthode Monte Carlo). La chaîne de calcul est basée sur quelques hypothèses qui constituent pour la plupart des paramètres ajustables ayant toutefois des valeurs spécifiées par défaut.

Enfin, le **quatrième chapitre** de cette thèse présente les premiers résultats obtenus avec la chaîne de calcul. Ces résultats sont comparés à des données de la littérature de manière à tester la cohérence des différentes hypothèses intégrées dans la simulation. L'influence du nouveau matériau ADN sur ces résultats est également estimée et différentes valeurs sont attribuées aux paramètres associés aux hypothèses de manière à estimer leur impact sur les résultats finaux. Ces derniers correspondent à l'ensemble des dommages calculés lors de la simulation des trois étapes précédemment mentionnées. Toutefois, un type de dommage est particulièrement mis en avant, il s'agit des cassures double brin à l'ADN ("Double Strand Break", DSB) qui sont particulièrement critiques du point de vue de la survie cellulaire [[OLIVE, 1998](#)].

Références

BERNAL, M., M. BORDAGE, J. BROWN, M. DAVÍDKOVÁ, E. DELAGE, Z. EL BITAR, S. ENGER, Z. FRANCIS, S. GUATELLI, V. IVANCHENKO, M. KARAMITROS, I. KYRIAKOU, L. MAIGNE, S. MEYLAN, K. MURAKAMI, S. OKADA, H. PAYNO, Y. PERROT, I. PETROVIC, Q. PHAM, A. RISTIC-FIRA, T. SASAKI, V. ŠTĚPÁN, H. TRAN, C. VILLAGRASA et S. INCERTI. 2015, «Track structure modeling in liquid water: A review of the Geant4-DNA very low energy extension of the Geant4 Monte Carlo simulation toolkit», *Physica Medica*, vol. 31, n° 8, p. 861–874. [2](#)

- BOLUS, N. E. 2001, «Basic review of radiation biology and terminology», *Journal of nuclear medicine technology*, vol. 29, n° 2, p. 67–73. [1](#)
- FRIEDLAND, W., M. DINGFELDER, P. KUNDRÁT et P. JACOB. 2011, «Track structures, DNA targets and radiation effects in the biophysical Monte Carlo simulation code PARTRAC», *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*, vol. 711, n° 1-2, p. 28–40. [2](#)
- GROSSWENDT, B. 2002, «Formation of ionization clusters in nanometric structures of propane-based tissue-equivalent gas or liquid water by electrons and alpha-particles», *Radiation and Environmental Biophysics*, vol. 41, n° 2, p. 103–112. [2](#)
- INCERTI, S., G. BALDACCHINO, M. BERNAL, R. CAPRA, C. CHAMPION, Z. FRANCIS, P. GUÈYE, A. MANTERO, B. MASCIALINO, P. MORETTO et OTHERS. 2010, «The Geant4-DNA project», *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 1, n° 02, p. 157–178. [2](#)
- KRÄMER, M. 2009, «Swift ions in radiotherapy–Treatment planning with TRiP98», *Nuclear Instruments and Methods in Physics Research Section B : Beam Interactions with Materials and Atoms*, vol. 267, n° 6, p. 989–992. [1](#)
- NIKJOO, H., S. UEHARA, D. EMFIETZOGLOU et F. CUCINOTTA. 2006, «Track-structure codes in radiation research», *Radiation Measurements*, vol. 41, n° 9-10, p. 1052–1074. [2](#)
- OLIVE, P. L. 1998, «The Role of DNA Single- and Double-Strand Breaks in Cell Killing by Ionizing Radiation», *Radiation Research*, vol. 150, p. 42–51. [3](#)
- ROSSI, H. H. 1959, «Specification of Radiation Quality», *Radiation Research*, vol. 10, p. 522–531. [1](#)

Chapitre 1

Calcul des dommages radio-induits à l'ADN

« Si la science ne s'intéresse pas aux choses délirantes, elle risque fort de passer à côté de choses intéressantes. »

Antoine Labeyrie, astronome, 12 août
2000

Sommaire

1.1 Effets biologiques des rayonnements ionisants	7
1.1.1 Caractérisation physique du dépôt d'énergie	7
1.1.2 Impact des rayonnements ionisants au niveau cellulaire	9
1.1.3 Effets à l'échelle de l'individu	11
1.1.4 Objectif de ce travail de thèse	13
1.2 Transport de particules avec la méthode de Monte Carlo	14
1.2.1 Méthode de Monte Carlo appliquée au transport de particules	14
1.2.2 Le code de simulation Geant4	17
1.2.3 L'extension Geant4-DNA	23
1.2.4 Utiliser Geant4/Geant4-DNA	26
1.3 Simulation de la radiolyse de l'eau	29
1.3.1 Généralités	29
1.3.2 La chimie dans Geant4-DNA	30
1.3.3 Utiliser le module chimie de Geant4-DNA	37
1.4 L'ADN et les dommages radio-induits	38
1.4.1 Structure de l'ADN : du nucléotide à la fibre de chromatine	39
1.4.2 Distribution spatiale de la chromatine dans le noyau cellulaire	40
1.4.3 Les dommages à l'ADN	43
1.4.4 Modélisations de l'ADN pour le calcul des dommages radio-induits	46
Références	54

1.1 Effets biologiques des rayonnements ionisants

La découverte des rayons X et de la radioactivité au 19^{ème} siècle eut une influence significative sur de nombreux domaines d'activités tels que la recherche, l'industrie et la médecine. Dans le domaine médical, cet impact découla de l'observation des effets biologiques des rayonnements ionisants qui rendit possible leur utilisation à des fins diagnostiques et thérapeutiques. Ces utilisations ont donné naissance à de nouvelles spécialités médicales telles que la radiologie, la radiothérapie et la curiethérapie. Néanmoins, l'usage des rayonnements ionisants au sein de ces spécialités a également mis en évidence des effets délétères sur les cellules saines illustrant ainsi la nécessité de bien identifier, doser et cibler la délivrance d'un traitement basé sur les rayonnements ionisants. Au delà du domaine médical, la caractérisation des rayonnements en termes d'effets sur le vivant est rapidement apparue comme une nécessité dans l'ensemble des activités liées aux rayonnements ionisants : industrie du nucléaire, applications militaires, exploration spatiale etc. Cette section introduit un état de l'art sur la compréhension actuelle du lien entre la caractérisation physique du rayonnement ionisant et son effet biologique.

1.1.1 Caractérisation physique du dépôt d'énergie

La caractérisation physique des effets d'un rayonnement s'effectue au travers de celle des dépôts d'énergie induits dans la matière par ce même rayonnement. Des grandeurs physiques y sont dédiées, il s'agit principalement de la fluence, de la dose absorbée et de la dose transférée (KERMA, pour "Kinetic Energy Released in MATter").

$$\text{Fluence } (n_p.cm^{-2}): \phi = \frac{dN}{da} \quad (1.1)$$

La fluence ϕ (cf. équation 1.1) correspond au nombre de particule dN pénétrant dans une sphère dont la projection en deux dimensions donne un cercle de surface da . La fluence s'exprime en nombre de particules par unité de surface ($n_p.cm^{-2}$).

$$\text{Dose absorbée (Gy): } D = \frac{\epsilon}{m} \quad (1.2)$$

La dose absorbée (cf. équation 1.2) est l'énergie ϵ déposée dans la matière par unité de masse m . Elle est exprimée en Gy ($J.kg^{-1}$) et il s'agit d'une grandeur moyenne. En effet, la dose absorbée est généralement mesurée à l'aide d'un grand nombre de mesures effectuées dans un volume fini (correspondant à une masse finie) suffisamment grand pour négliger les fluctuations statistiques inhérentes à cette mesure.

$$\text{Dose transférée (Gy): } K = \frac{E_{tr}}{m} \quad (1.3)$$

La dose transférée (KERMA, cf. équation 1.3) est exprimée dans la même unité que la dose absorbée, c'est à dire le Gy. Cette grandeur correspond à l'énergie cinétique transférée par un rayonnement non directement ionisant à des particules directement ionisantes lors de son passage dans la matière. Les rayonnements indirectement ionisants sont, par exemple, ceux constitués de photons ou de neutrons. La dose transférée est une grandeur moyenne par nature, comme la dose absorbée.

La nature statistique de la dose absorbée et de la dose transférée est liée à la façon dont l'énergie ϵ est déposée dans la matière. En effet, l'énergie moyenne déposée dans un

volume donné de masse m provient d'un ensemble d'interactions individuelles particule-matière à l'origine de nombreux dépôts d'énergie individuels ε_i . Ces dépôts individuels sont quantitativement et qualitativement divers, reflétant ainsi la diversité des interactions physiques pouvant avoir lieu dans la matière (ionisation, excitation ...) et le caractère aléatoire de leur occurrence. La figure 1.1 représente l'évolution du rapport $\frac{\varepsilon}{m}$ en fonction de la masse. Sur cette figure, il existe un seuil, associé à une masse m_1 , en deçà duquel la valeur de ce ratio est instable tandis qu'au delà duquel elle est stable. Ceci signifie que, pour des volumes de masses suffisantes ($> m_1$), les dépôts d'énergie formant ε sont suffisamment nombreux pour que leur nature stochastique ne transparaisse pas. Dans ce cas, la dose absorbée est une bonne représentation de l'énergie déposée. A l'inverse, avec des volumes de faibles masses ($< m_1$) les dépôts ne sont pas suffisamment nombreux et ne peuvent s'équilibrer : leur nature stochastique devient visible et la valeur de ε varie fortement d'un volume à l'autre. Les valeurs des rapports sont constantes entre les valeurs m_1 et m_2 mais, par contre, décroissent lorsque m devient supérieur à m_2 . Ceci s'explique par le fait que les rayonnements traversant des volumes de masses importantes sont progressivement atténués et absorbés. Au final, la dose absorbée est une grandeur pertinente pour des volumes ayant des masses comprises entre m_1 et m_2 , c'est à dire des volumes où les interactions sont en nombre suffisant pour que l'énergie déposée soit statistiquement stable d'un volume à un autre. Lorsque les volumes ou masses sont suffisamment pe-

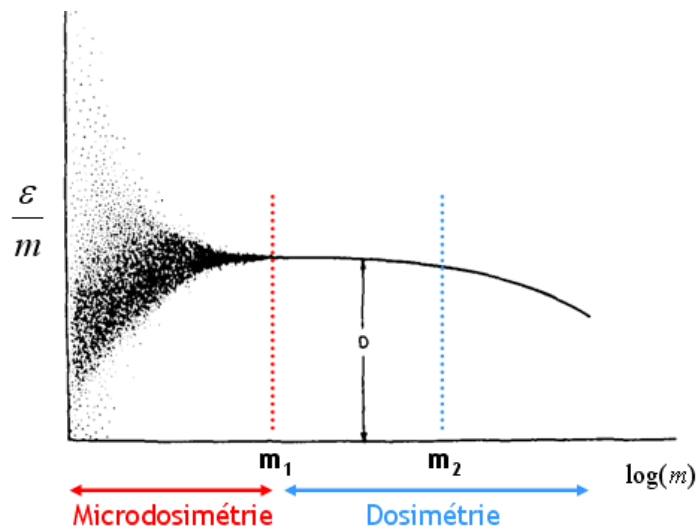


FIGURE 1.1 – Variation du rapport $\frac{\varepsilon}{m}$ en fonction de la masse m du site étudié [ROSSI, 1968].

tits pour que la nature stochastiques des dépôts devienne dominante, c'est-à-dire pour des masses inférieures à m_1 , alors la nature stochastique de la dose absorbée ne peut plus être négligée : il s'agit alors du domaine de la **microdosimétrie** [ROSSI, 1959]. La microdosimétrie se caractérise par le fait d'utiliser un volume de taille micrométrique au moyen de grandeurs physiques ayant un comportement stochastique. Parmi ces grandeurs se trouve notamment l'énergie spécifique z exprimée en Gy qui est un l'équivalent stochastique de la dose absorbée. L'énergie spécifique est définie par l'équation 1.4 comme le rapport entre une quantité d'énergie ε déposée dans un volume et la masse m de celui-ci.

$$z = \frac{\varepsilon}{m} = \frac{\sum \varepsilon_i}{m} \quad (1.4)$$

Il s'agit d'une définition similaire à celle de la dose absorbée à ceci près que la dose absorbée est une variable *moyenne* tandis que l'énergie spécifique est une variable *stochastique*. En d'autres termes, la dose absorbée est représentée par une valeur fixe dans une

situation donnée tandis que l'énergie spécifique varie aléatoirement. En raison des variations de l'énergie spécifique, il est possible d'y associer une densité de probabilité f . Finalement, la moyenne des différentes valeurs prises par l'énergie spécifique permet de faire le lien avec la dose absorbée comme l'explique l'équation 1.5.

$$D = \bar{z} = \int_0^{z_{max}} z \cdot f(z) dz \approx \frac{\sum_{i=0}^{i=n} z_i}{n} \quad (1.5)$$

avec D la dose absorbée, \bar{z} l'énergie spécifique moyenne, z l'énergie spécifique et n le nombre de "mesures" de z . Précisons que les dépôts d'énergie individuels à l'origine des valeurs prises par l'énergie spécifique et la dose absorbée sont distants entre eux de quelques nanomètres. Cette distance étant inférieure au micromètre de plusieurs ordres de grandeurs, la microdosimétrie ne permet pas d'accéder directement à ces dépôts.

Une autre grandeur utilisée pour caractériser les rayonnements est le Transfert d'Énergie Linéique (TEL). Le TEL ($keV.um^{-1}$) décrit l'énergie transférée par une particule ionisante à la matière dE_{Δ} par unité de distance dx tout en excluant l'énergie déposée les particules secondaires ayant une énergie cinétique supérieure à Δ .

$$TEL_{\Delta} = \frac{dE_{\Delta}}{dx} \quad (1.6)$$

Le TEL a pour avantage de ne pas dépendre uniquement de la quantité d'énergie totale déposée par une particule dans un milieu mais également de la façon dont cette énergie est déposée le long du parcours d'une particule. Cependant, le TEL moyenne (ou néglige avec un faible Δ) l'énergie déposée radialement le long du parcours de la particule incidente comme illustré en figure 1.2 où deux particules incidentes ont un même TEL ("a" et "c") malgré une répartition radiale des dépôts d'énergie différente. En effet, les dépôts sont plus proches du cœur de la trace (points de couleur rose) en "c" qu'en "a" mais cette distinction ne se retrouve pas dans la valeur du TEL associée à "a" et "c" qui est identique.

Finalement, la grandeur physique servant de base à la caractérisation des effets des rayonnements sur le vivant est généralement la dose absorbée. Et ceci malgré le fait qu'elle ne rende pas compte de la répartition spatiale des dépôts d'énergies dans la matière. Parmi les différentes raisons à l'origine de son utilisation se trouvent certainement la relative facilité avec laquelle elle est mesurable comparativement aux grandeurs utilisées en microdosimétrie par exemple. La prise en compte des différents types d'irradiation, liés aux répartitions spatiales des dépôts d'énergie différentes en fonction des rayonnements et du milieu considéré, s'effectue toutefois au moyen du RBE ("Relative Biological Effectiveness") qui relie la dose à un effet biologique relatif. Ce concept sera abordé avec plus de détails par la suite.

1.1.2 Impact des rayonnements ionisants au niveau cellulaire

Pour améliorer la compréhension des effets biologiques des rayonnements ionisants, diverses recherches sont menées et, notamment, des recherches en biologie se focalisent sur les effets à l'échelle de cellule. Lorsque qu'une cellule est exposée à des rayonnements ionisants, elle peut être endommagée et cela a généralement trois conséquences directement observables par les biologistes [BOLUS, 2001] et facilement accessibles expérimentalement :

- La division cellulaire est retardée

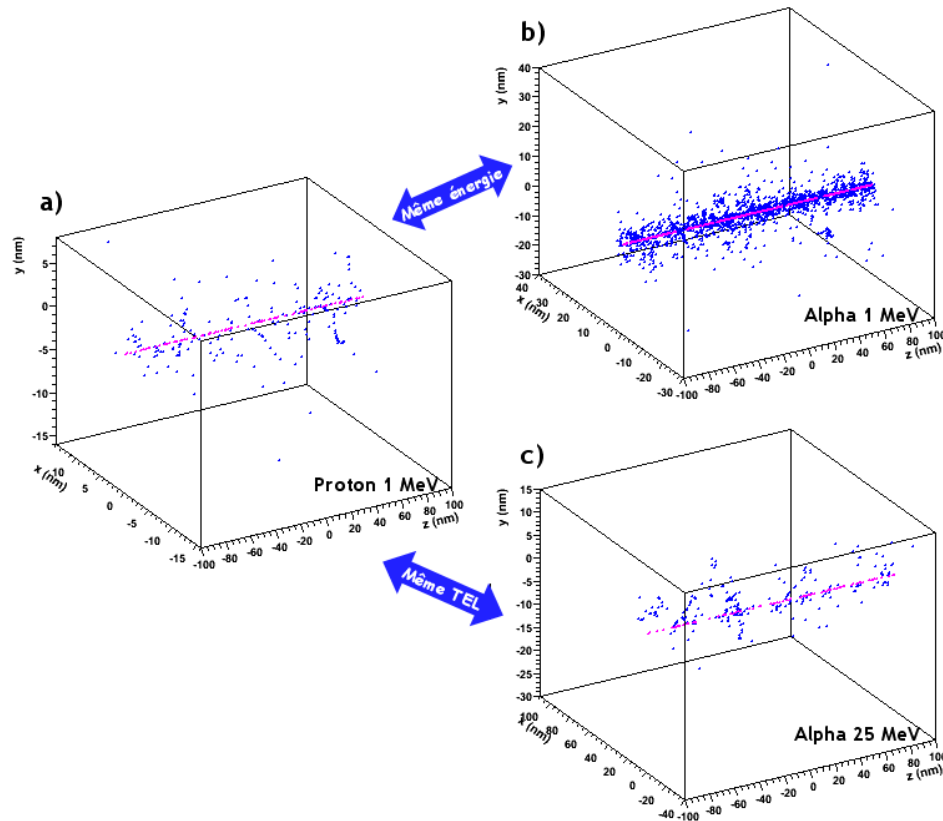


FIGURE 1.2 – Les dépôts d'énergie de protons (1 MeV) et alphas (1 MeV et 25 MeV) obtenus avec Geant4-DNA. En rose sont représentés les dépôts d'énergie de la particule primaire et en bleu les dépôts d'énergies issus des particules secondaires. [DOS SANTOS, 2013]

- L'arrêt de la division cellulaire
- La mort de la cellule

Ces effets discernables ont permis la réalisation d'études empiriques sur la survie cellulaire qui ont débouché sur la création du concept de RBE ("Relative Biological Effectiveness"). Le RBE est une valeur rendant compte de l'effet biologique d'un type de rayonnement non pas de manière absolue mais *relativement* à un autre rayonnement. Le RBE est défini comme étant le rapport de la dose absorbée provenant d'un rayonnement "standard" par rapport à celle provenant d'un autre rayonnement lorsque ces deux doses induisent un même effet biologique. Le RBE est calculé à l'aide de l'équation 1.7 pour un effet et un système biologique donnés tels que, par exemple, la survie de cellules d'endothélium.

$$\text{RBE} = \frac{D_{ref}}{D_{ray}}, \text{ pour un effet et un système biologique donnés} \quad (1.7)$$

avec D_{ref} la dose absorbée par un rayonnement de référence et D_{ray} la dose absorbée par un rayonnement d'intérêt. La figure 1.3 montre la variation de la survie cellulaire d'un même type de cellule en fonction de la dose absorbée et pour deux rayonnements distincts : des photons et des ions carbone. Si la survie cellulaire à 10% est choisie comme effet biologique pour le calcul du RBE (RBE_{10}) alors la figure 1.3 illustre le fait que deux valeurs de dose sont obtenues pour un même effet : 7 Gy pour les photons et 4 Gy pour les ions carbone de 266,4 MeV. Ces deux valeurs permettent de calculer le RBE associé à cette survie cellulaire et ces rayonnements (photons et ions carbone), soit : $\text{RBE}_{10} = 1,6$. La figure 1.3 montre également que pour un rayonnement (le carbone ^{12}C), une dose absorbée, un effet et un système biologique fixé, le RBE n'est pas constant et varie en fonction

de l'énergie du rayonnement. Par exemple, pour une dose de 2 Gy, les ions carbone de 11 MeV ont un RBE de 4,2 tandis que celui des ions de 266,4 MeV vaut 6,6. Ces variations sont dues au fait que le RBE est basé sur le concept de dose absorbée qui ne prend pas en compte la répartition spatiale des dépôts d'énergie dans la matière. Au final, le RBE est une grandeur relative qui ne permet pas de caractériser à elle seule l'effet d'un rayonnement.

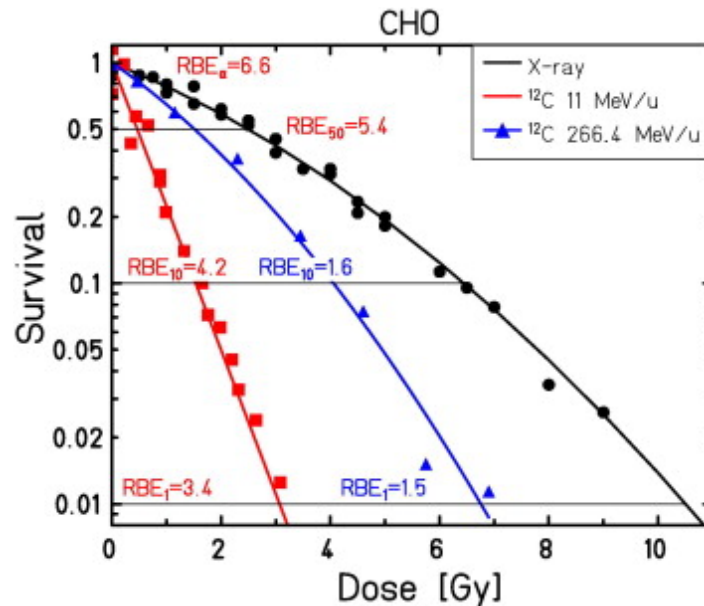


FIGURE 1.3 – La survie cellulaire en fonction de la dose absorbée obtenue suite à un même type de cellules (CHO) pour deux rayonnements distincts : des photons (bleu) et des ions carbone (rouge) [KRÄMER, 2009].

1.1.3 Effets à l'échelle de l'individu

L'absorption de rayonnements ionisants par des tissus biologiques provoque des effets variés à l'échelle d'un individu. Ces effets sont généralement classés en deux catégories principales : les effets déterministes et les effets stochastiques. Les **effets déterministes** sont prédictibles et dépendent de nombreux facteurs tels que l'âge d'un individu, la dose absorbée, la durée de l'exposition etc. Ils se produisent suite à l'absorption de fortes doses (> 2 Sv) et incluent, entre autres, le syndrome hématopoïétique, les érythèmes, le syndrome gastro-intestinal etc. Ces effets n'apparaissent jamais en deçà d'un seuil S_{min} mais toujours au delà d'un seuil S_{max} . Enfin, les effets déterministes augmentent en intensité parallèlement à la dose reçue. A l'inverse, les **effets stochastiques** apparaissent toujours de manière aléatoire mais leur probabilité d'occurrence croît avec la dose reçue. Notons cependant que si la probabilité d'occurrence d'un effet stochastique croît avec la dose absorbée, ce n'est pas le cas de sa gravité qui en est totalement indépendante. Il est communément admis que la carcinogenèse et les mutations héréditaires font partie des effets stochastiques [DOWD et TILSON, 1999]. La distinction entre les effets déterministes et les effets stochastiques [BOLUS, 2001] est particulièrement utilisée pour l'étude des effets des rayonnements ionisants car elle correspond à une différence d'approche. En effet, les effets déterministes sont aisément accessibles expérimentalement contrairement aux effets stochastiques qui requièrent d'importantes études épidémiologiques. Parallèlement aux effets déterministes ou stochastiques, une autre distinction peut être effectuée

en se basant sur l'intervalle de temps durant lequel une dose donnée est absorbée par un individu. Cette distinction implique les notions d'**irradiation aiguë** et d'**irradiation chronique**. Une irradiation aiguë se caractérise par un temps d'exposition court à l'inverse d'une irradiation chronique. A faible dose, une irradiation aiguë est cliniquement peu visible car les premiers effets à l'échelle d'un individu apparaissent autour de 1 Sv [LADOU, 1990]. A de fortes doses, les effets les plus immédiats associés à l'irradiation sont de nature déterministes et caractéristiques du syndrome d'irradiation aiguë [DOWD et TILSON, 1999; STROM, 2003]. Le tableau 1.1 indique les principales formes que peut prendre le syndrome d'irradiation aiguë. Une irradiation chronique à de fortes doses est une situation rare car un syndrome d'irradiation aiguë (déterministe) apparaît rapidement et généralement est suivi de mesures destinées à faire cesser l'irradiation. Une irradiation chronique associée à de faibles doses est une situation plus fréquente car les effets en sont majoritairement stochastiques et donc peu visibles. Ces effets stochastiques englobent l'apparition de cancers [LADOU, 1990].

TABLEAU 1.1 – Le syndrome d'irradiation aiguë [STROM, 2003]

Forme du syndrome	Exposition (Gy)
Hématopoïétique	> 1-2 Gy
Cutanée	> 4-7 Gy
Pulmonaire	> 6-9 Gy
Gastro-intestinale	> 6-10 Gy
Cérébro-vasculaire	> 20-50 Gy

D'un point de vue prédictif, les effets déterministes sont relativement aisés à prendre en compte car leurs conditions d'apparition sont bien identifiées. La radiothérapie et la curiethérapie sont des domaines médicaux s'appuyant précisément sur des effets déterministes identifiés et liés à la sur-sensibilité des tissus de tumeurs comparée à celle des tissus sains. En effet, d'une manière très générale, il est possible de corrélérer la sensibilité d'un tissu biologique à la vitesse avec laquelle ses cellules se renouvellent : les tissus dont les cellules se renouvellent rapidement sont les plus sensibles tandis que ceux dont les cellules se renouvellent plus rarement présentent une meilleure résistance. Les cellules cancéreuses se divisant généralement plus rapidement que les cellules saines, les tissus cancéreux présentent donc une sensibilité aux rayonnements supérieure à celle des tissus sains, illustrant ainsi l'intérêt thérapeutique des rayonnements ionisants. Contrairement aux effets déterministes, les effets stochastiques sont difficiles à observer et leurs probabilités d'occurrence lors d'exposition à de faibles doses sont encore mal connues. En effet, les données disponibles doivent être très nombreuses pour obtenir des résultats statistiquement significatifs dans un contexte épidémiologique. En général, de telles données sont collectées suite à des accidents tels que ceux de Tchernobyl [MOYSICH et al., 2002] et Fukushima [BOICE JR, 2012] ou suite à des événements tels que les bombardements de Hiroshima et Nagasaki [PIERCE et PRESTON, 2000]. En conséquence, il n'existe pas de modèle prédictif faisant consensus au sujet du risque associé à une irradiation à de faibles doses. Un tel modèle aurait pourtant des applications potentielles dans de nombreux domaines et pourrait permettre d'améliorer la connaissance du risque encouru par la population en général ou, plus spécifiquement, par les astronautes suite à leur exposition à des radiations lors de missions hors de la zone protégée par le champ magnétique terrestre [CUCINOTTA et al., 2001; TOWNSEND, 2005]. Les éventuelles complications suite à des traitements ou examens médicaux impliquant des radiations pourraient également être mieux évaluées. Cependant, les quelques modèles prédictifs actuellement dévelop-

pés sont très spécifiques et ont un domaine de validité restreint en raison principalement des limites associées aux données épidémiologiques dont ils sont issus. Finalement, leurs domaines de validité restreints rendent leur utilisation hors "conditions de développement" problématiques limitant ainsi les applications potentielles et leur fiabilité.

1.1.4 Objectif de ce travail de thèse

L'objectif de ce travail de thèse est de développer un modèle mécanistique multi-échelle capable de simuler les dommages radio-induits précoces à l'ADN dans le cadre d'irradiations effectuées avec des protons et des ions (alpha). La construction d'une telle modélisation vise à fournir un outil de simulation capable de tester des hypothèses autrement que par l'expérimentation et fait partie d'un projet plus général mené à l'IRSN : le projet ROSIRIS. Ce dernier a pour finalité la construction d'un modèle prédictif du risque associé à une exposition médicale. Comme mentionné en section 1.1.3, un tel outil serait particulièrement intéressant dans un cadre médical et également dans le cadre des faibles doses où les données expérimentales sont particulièrement difficile d'accès. La simulation prise en charge par cet outil est dédiée au calcul des dommages radio-induits précoces à l'ADN et débute par le transport de particules dans la matière. Le transport de ces particules vise à reproduire une répartition spatiale réaliste des dépôts d'énergie dans la matière. L'importance de cette répartition spatiale est apparue avec la prise en compte de cibles spécifiques particulièrement sensibles dans les simulations de ce type. Il s'agit par exemple du noyau cellulaire ou d'éléments contenus par ce dernier : l'ADN entre autres. C'est la prise en compte de cette répartition ainsi que de cibles de l'ordre du micromètre qui a donné naissance, dans un premier temps, à la micro-dosimétrie. La considération de cibles encore plus spécifiques, de l'ordre du nanomètre, telle que la molécule d'ADN, a entraîné le besoin d'aller vers l'étude des interactions entre molécules et donné naissance à la **nano-dosimétrie**. Cette dernière prend en compte l'ensemble des dépôts d'énergie et inclut ainsi la totalité de leur répartition spatiale. La simulation développée dans ce travail est basée sur une approche mécanistique, c'est à dire que l'on essaye de reproduire par la simulation les dommages précoces à l'ADN occasionnés par l'irradiation d'une cellule. Cette approche requiert d'incorporer de très nombreux paramètres et de construire une simulation particulièrement complexe. Parmi ces paramètres, nombreux sont ceux qui doivent avoir été préalablement mesurés ou calculés.

D'un point de vue mécanistique, l'irradiation d'une cellule va déclencher une chaîne d'événements généralement regroupés dans de grandes étapes temporellement distinctes : les étapes physique, physico-chimique, chimique, bio-chimique et biologique. La figure 1.4 présente ces différents étapes ainsi qu'une partie des événements associés. Lorsqu'elle est irradiée, une cellule peut subir deux types de dommages communément classés en dommages directs et indirects. Les **dommages directs** sont provoqués par les particules mises en mouvement lors de l'irradiation. En effet, ces dernières peuvent interagir avec différents éléments situés à l'intérieur de la cellule (membrane, protéines, ADN etc.) et les endommager. Les **dommages indirects** sont causés par les espèces chimiques générées suite aux interactions entre les particules et les molécules d'eau de la cellule. En effet, certaines molécules d'eau ionisées ou excitées se dissocient et génèrent diverses espèces chimiques telles que des radicaux libres. Ces espèces chimiques interagissent également avec les différents éléments biologiques de la cellule et, ce faisant, peuvent les endommager. Les dommages précoces occasionnés à la cellule ont des conséquences variées. Par exemple, dans le cas de dommages à l'ADN, un dommage isolé ("simple") sera très probablement réparé par la cellule et donc sans conséquence. A l'inverse, deux dom-

mages suffisamment proches et situés sur deux brins distincts constitueront une cassure dite double-brin, dommage particulièrement critique [OLIVE, 1998]. En effet, une cassure double-brin a de fortes chances de causer une mutation ou encore d'induire la mort de la cellule concernée. Finalement, à l'échelle d'une cellule, seuls certains dommages sont observables expérimentalement car ils déclenchent un effet visible. Notons qu'il en est de même à l'échelle d'un être humain car la mutation et la mort de cellules n'a pas toujours le même impact : certaines cellules sont plus facilement remplaçables que d'autres.

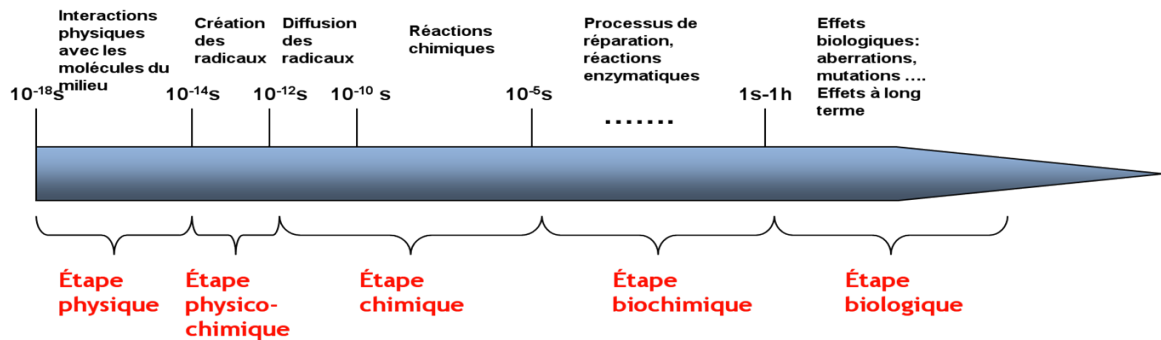


FIGURE 1.4 – Les effets radio-induits repartis en cinq grandes étapes temporellement distinctes.

La construction d'une simulation mécanistique pour le calcul des dommages radio-induits sur l'ADN nécessite la prise en compte de chacune des grandes étapes listées en figure 1.4. Parmi ces dernières, la simulation de l'étape physique implique de modéliser le trajet de particules dans la matière ainsi que les interactions particule/matière à l'échelle nanométrique, et la simulation des étapes physico-chimique et chimique revient à modéliser la radiolyse de l'eau ainsi que la diffusion des espèces chimiques générées à cette occasion et leurs interactions avec l'ADN. Les sections 1.2 et 1.3 présentent l'utilisation de la méthode de Monte Carlo pour la simulation de ces trois étapes. Plus précisément, la section 1.2 traite des interactions physiques des particules dans la matière et la section 1.3 de la simulation de la radiolyse de l'eau. Enfin, la section 1.4 introduit la molécule d'ADN ainsi que les différentes manières de la modéliser.

1.2 Transport de particules avec la méthode de Monte Carlo

1.2.1 Méthode de Monte Carlo appliquée au transport de particules

La méthode de Monte Carlo [METROPOLIS, 1987; METROPOLIS et ULAM, 1949] est une méthode de calcul numérique basée sur des tirages aléatoires. Une succession de ces tirages est réalisée de manière à échantillonner les variables aléatoires du problème traité pour en déduire une valeur d'intérêt. Répété plusieurs fois, ce processus permet d'obtenir une distribution des valeurs d'intérêt et donc une estimation de leur moyenne et de leur intervalle de confiance associé. Toutefois, le nombre d'échantillons doit être suffisamment important pour que la moyenne empirique des résultats soit un estimateur sans biais de l'espérance de la grandeur d'intérêt (cf. théorème 1) et que leur distribution soit telle que prédite par le théorème central-limite (cf. théorème 2). La méthode de Monte Carlo est aujourd'hui utilisée dans de nombreux domaines [KROESE et al., 2014] et particulièrement en physique pour modéliser les interactions entre des particules et un milieu donné. Différents codes [NIKJOO et al., 2006], dits "codes de Monte Carlo", permettent de réaliser ces modélisations. Parmi ceux-ci, citons notamment les codes MCNP

[GOORLEY et al., 2012], PENELOPE [SALVAT et al., 2009], TRIPOLI [BRUN et al., 2015] et GEANT4 [AGOSTINELLI et al., 2003; ALLISON et al., 2006].

Théorème 1. (Loi des grands nombres) Soit $\{X_i\}_{i \in \mathbf{N}}$ une suite de variables aléatoires indépendantes et identiquement distribuées, d'espérance μ . On pose : $\forall N \in \mathbf{N}, Y_N = \frac{1}{N} \sum_{i=1}^N X_i$. Alors la suite $\{Y_i\}_{i \in \mathbf{N}}$ converge presque sûrement vers μ [DIEUDONNÉ, 2013].

Théorème 2. (Central-limite) Soit $\{X_i\}_{i \in \mathbf{N}}$ une suite de variables aléatoires indépendantes et identiquement distribuées, d'espérance μ et de variable finie σ^2 . On pose : $\forall N \in \mathbf{N}, Y_N = \frac{1}{N} \sum_{i=1}^N X_i$ et $\forall N \in \mathbf{N}, Z_N = \frac{Y_N - \mu}{\sigma \sqrt{N}}$. Alors la suite $\{Z_i\}_{i \in \mathbf{N}}$ converge en loi vers la loi normale centrée [DIEUDONNÉ, 2013].

Les codes de Monte Carlo dédiés au transport de particules dans la matière fonctionnent selon un modèle relativement constant et basé sur la notion de **section efficace** (σ) exprimée en barn (b) ($1 \text{ barn} = 10^{-28} \text{ m}^2$). Cette section efficace est une grandeur physique représentative de la probabilité de collision entre une particule incidente et une particule cible. Ainsi, dans le cas simple d'une surface S prise pour cible par une particule incidente et contenant cinq atomes représentés par des points uniformément distribués, alors la section efficace de chaque atome correspond à un disque de surface (σ) centré sur la position de l'atome considéré. Ceci est illustré en figure 1.5. Si les points d'impact des particules incidentes avec la surface S sont distribués aléatoirement sur cette dernière alors un point d'impact localisé sur la surface occupée par un disque "section efficace" sera considéré comme le point d'occurrence d'une interaction entre la particule incidente et l'atome considéré. A partir de ce concept de section efficace, il est possible de définir

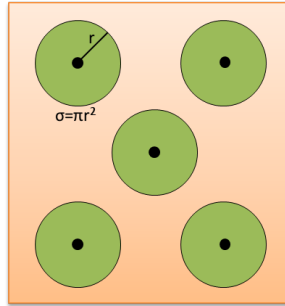


FIGURE 1.5 – Représentation graphique des sections efficaces σ (disques verts) de 5 atomes (points noirs) situés sur une surface S (orange).

le **libre parcours moyen** (λ) d'une particule au moyen de l'équation 1.8 :

$$\lambda = \frac{1}{N\sigma} \quad (1.8)$$

avec λ le libre parcours moyen, σ la section efficace associée aux interactions particule/-cible et N le nombre de cibles par unité de volume dans le milieu considéré. Ce libre parcours moyen correspond à la valeur moyenne d'une variable aléatoire associée au chemin parcouru par une particule entre deux interactions avec le milieu considéré. A l'aide du libre parcours moyen, il est possible d'établir une densité de probabilité correspondant à la distance parcourue entre deux interactions :

$$p(s) = \frac{1}{\lambda} e^{-s/\lambda} \quad (1.9)$$

où s est la distance parcourue entre deux interactions et $p(s)$ la probabilité qu'une particule parcourt effectivement cette distance.

Démonstration. Une façon de retrouver l'équation 1.9 consiste à définir la probabilité p_{ds} de rencontrer une cible si la particule parcourt une petite distance ds comme :

$$p_{ds} = \frac{ds}{\lambda} \quad (1.10)$$

et la probabilité $F(s)$ de parcourir une distance s sans interagir comme :

$$F(s) = 1 - \int_0^s p(u) du = \int_s^\infty p(u) du \quad (1.11)$$

La probabilité $p(s)ds$ de parcourir une distance s sans interagir puis d'interagir entre s et $s + ds$ est alors le produit des équations 1.10 et 1.11, soit :

$$\begin{aligned} p(s)ds &= F(s)p_{ds} \\ p(s)ds &= \frac{ds}{\lambda} \int_s^\infty p(u) du \end{aligned}$$

Ce qui équivaut à écrire :

$$p(s) = \frac{1}{\lambda} \int_s^\infty p(u) du$$

En dérivant par rapport à s , on obtient :

$$\frac{dp(s)}{ds} = -\frac{1}{\lambda} p(s)$$

D'où on déduit :

$$p(s) = Ae^{-s/\lambda} \quad (1.12)$$

avec A une constante déterminée par la normalisation $\int_0^\infty p(s) ds = 1$, soit :

$$\int_0^\infty p(s) ds = 1 \Leftrightarrow A = \frac{1}{\lambda}$$

En remplaçant A par son expression dans l'équation 1.12, on obtient finalement l'équation 1.9 :

$$p(s) = \frac{1}{\lambda} e^{-s/\lambda}$$

□

Dans une simulation dédiée au transport de particules, la densité de probabilité spécifiée par l'équation 1.9 permet d'échantillonner la distance parcourue par une particule entre deux interactions. Pour effectuer cet échantillonnage, il est usuel de calculer la fonction de partition $F(s)$ correspondant à la densité de probabilité $p(s)$ étudiée. La méthode de transformation de variable est ensuite utilisée pour lier une variable n uniformément distribuée entre 0 et 1 et la variable aléatoire correspondant à la distance parcourue. $F(s)$ est généralement exprimée de la façon suivante :

$$\begin{aligned} F(s) &= \frac{1}{\lambda} \int_0^s e^{-u/\lambda} du \\ F(s) &= 1 - e^{-s/\lambda} \end{aligned}$$

et comme :

$$n = F(s)$$

Nous avons donc :

$$n = 1 - e^{-s/\lambda} \Leftrightarrow s = -\lambda \ln(1 - n)$$

Ainsi, le tirage aléatoire de nombres n_i uniformément repartis sur $[0, 1]$ ensuite introduits dans l'équation 1.13, permet d'obtenir des nombres s_i répartis selon la densité de probabilité $p(s)$ (cf. équation 1.9).

$$s_i = -\lambda \ln(1 - n_i) \tag{1.13}$$

En pratique, l'échantillonnage de la distance parcourue par une particule entre deux interactions s'effectue au sein d'un algorithme prenant en compte le type d'interaction rencontrée et son influence sur la particule "transportée". Une version simplifiée du fonctionnement d'un tel algorithme est présenté avec l'algorithme 1.1 mais, en pratique, chaque code de Monte Carlo implémente une variante. Parmi toutes les variations possibles, le fait d'utiliser, ou non, une technique dite d'**histoire condensée** [LARSEN, 1992] pour accélérer la simulation a une importance particulière dans ce travail. En effet, avec des histoires condensées, seules les interactions les plus significatives pour la simulation sont véritablement simulées. Ces interactions sont généralement celles déposant le plus d'énergie dans le milieu. Les multiples interactions "ignorées" dans une telle simulation sont cependant prises en compte au moyen d'une perte d'énergie continue et distribuée le long du trajet parcouru entre deux interactions "significatives". Les codes fonctionnant avec ce modèle sont particulièrement rapides mais ne peuvent fournir une cartographie précise de l'ensemble des interactions ayant eu lieu dans un volume. À l'inverse, un code ignorant cette technique sera plus lent mais donnera accès à plus d'interactions et donc à une répartition spatiale plus détaillée [BALLINGER et al., 1992]. Un tel code est communément appelé un **code de structure de trace** car il donne accès à la structure spatiale de la trace, c'est à dire à l'ensemble des positions des interactions générées par le transport d'une particule dans la simulation. L'approche utilisée est alors qualifiée d'approche "pas à pas". Notons que l'ensemble des codes contiennent un seuil en énergie en deçà duquel les particules ne sont plus transportées. Le positionnement de ce seuil influe grandement sur la précision de la cartographie spatiale des interactions. Généralement, les codes qui utilisent une approche "histoire condensée" ont un seuil en énergie supérieur à ceux mettant en œuvre une approche "pas à pas". Ceci contribue également à améliorer la précision de la cartographie spatiale des interactions générées par les codes "pas à pas". La figure 1.6a liste des codes de Monte Carlo utilisant les histoires condensées tandis que la figure 1.6b liste des codes dits "de structures de trace". Geant4-DNA est aussi un code de structure de trace bien qu'il ne soit pas présent sur cette figure produite peu avant la première distribution officielle de Geant4-DNA.

Les codes de structure de trace sont particulièrement adaptés à l'étude des dommages radio-induits à l'ADN car les constituants de ce dernier ont des dimensions de l'ordre du nanomètre. En effet, une telle taille de cibles impose une résolution spatiale similaire en ce qui concerne la localisation des interactions des particules transportées. Et, pour les raisons précédemment mentionnées, une telle précision ne peut être atteinte qu'avec un code de structure de trace.

1.2.2 Le code de simulation Geant4

Le code de Monte Carlo Geant4 (GEometry ANd Tracking version 4) [AGOSTINELLI et al., 2003; ALLISON et al., 2006] est un code écrit en C++ (programmation orientée objet) qui se présente sous la forme d'une boîte à outils ("bibliothèque" en informatique) contenant les éléments nécessaires pour effectuer du transport de particules. Une collaboration internationale d'une centaine de chercheurs et ingénieurs se charge de développer et main-

A list of Monte Carlo radiation transport codes

Code	Particle	Medium	Energy range ^a	Reference
ETRAN	e ⁻ and photon	All	10 keV–1 GeV	Berger and Seltzer (1973)
EGS4	e ⁻ and photon	All	10 keV–1 GeV	Nelson et al. (1985)
FLUKA	p,n, meson	All	1 keV–GeV	Fasso et al. (2005)
GEANT4	p,n, meson	All	250 eV–GeV	Agostinelli et al. (2003)
MCEP	e ⁻ photon	All	1 keV–30 MeV	Uehara (1986)
MCNP5	n, photon, e ⁻	All	See ref.	Goorley et al. (2003)
MCNPX	n, light ions	All	See ref.	Hendricks et al. (2005)
PENELOPE	e ⁻ and e ⁺	All	100 eV–1 GeV	Salvat et al. (2003)
PHITS	HZE	All	MeV–GeV	Iwase et al. (2002)
PEREGRINE	e ⁻ and photon	All	Therapy beams	Hartmann Siantar and Moses (1998)
PTRAN	Protons	Water	<250 MeV	Berger (1993)
SRIM	All ions	All	keV–2 GeV/u	Ziegler et al. (2003)
SHIELD-HIT	1 < Z < 10	All	1 MeV/u–1 TeV/u	Gudowska et al. (2004)

^aThese parameters may change as the codes are in continuous state of evolution and improvement. Readers should see the URL for the codes listed in the reference list.

(a) Codes utilisant l'approche d'histoire condensée

A list of Monte Carlo track-structure codes suitable for biophysical modeling at molecular level

Code	Particle	Energy range	Cross-section database ^a	Reference
CPA100 ^{b,c}	e ⁻	≥ 10 eV–100 eV e ⁻	Wat. (l)	Terrissol and Beaudre, 1990
DELTA ^c	e ⁻	≥ 10 eV–10 keV e ⁻	Wat. (v)	Zaider et al. (1983)
ETRACK ^c	e ⁻ , p, α	≥ 10 eV–10 keV e ⁻	Wat. (v)	Ito (1987)
KURBUC ^c	e ⁻	≥ 10 eV–10 MeV e ⁻	Wat. (v)	Uehara et al. (1993)
LEEPS	e ⁻ , e ⁺	0.1–100 keV	Many materials	Fernandez-Varea et al. (1996)
LEPHIST ^c	p	≥ 1 keV–1 MeV	Wat. (v)	Uehara et al. (1993)
LEAHIST ^c	α	≥ 1 keV/u–2 MeV/u	Wat. (v)	Uehara and Nikjoo (2002a)
MC4	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Emfietzoglou et al. (2003)
NOTRE DAME ^c	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Pimblott et al. (1990)
OREC ^c	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Turner et al. (1983)
PARTRAC ^{b,c}	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Friedland et al. (2003)
PITS04 ^b	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (l)	Wilson et al. (2004)
PITS99 ^c	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v)	Wilson and Nikjoo (1999)
SHERBROOKE ^c	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Cobut et al. (2004)
STBRGEN ^c	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Chatterjee and Holley (1993)
TRION	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Lappa et al. (1993)
TRACEL ^c	e ⁻ , ions	≥ 10 eV e ⁻ , ions ≥ 0.3 MeV/u	Wat. (v,l)	Tomita et al. (1997)

^aNomenclatures 'l' and 'v' have been used for liquid and vapor. In reality it is not easy to distinguish between these modes as experimental cross-sections for water have been measured only in water vapor or ice phase (see Section 7).

^bThese codes have implemented the theoretical model of liquid water by Dingfelder et al. (1998).

^cThese codes have extension for generating distribution of radicals at 10⁻¹² s and later times.

(b) Codes dits "de structure de trace"

FIGURE 1.6 – Liste de codes de Monte Carlo [NIKJOO et al., 2006].

Données : Position et direction d'une particule primaire

Résultat : Le transport de la particule primaire est effectué

1. Position P et direction D initiales d'une particule primaire d'énergie E dans un milieu homogène de densité N.
2. Quelles sont les interactions particule/milieu possibles ? Par exemple : ionisation avec ($\sigma_{ioni}(E)$), excitation ($\sigma_{exci}(E)$) et diffusion élastique ($\sigma_{diff}(E)$).
3. Calculer le libre parcours moyen correspondant à chacune des interactions possibles en utilisant l'équation 1.8.
4. Échantillonner une distance pour chacune des interactions à l'aide de l'équation 1.13 et de trois tirages aléatoires uniformes sur [0, 1] pour obtenir :

$$\begin{cases} s_{ioni} \\ s_{exci} \\ s_{diff} \end{cases}$$

5. Choisir la plus petite distance s_{min} parmi s_{ioni} , s_{exci} et s_{diff} pour sélectionner la prochaine interaction.
6. Déterminer la nouvelle position P_n à partir de la position P à laquelle est ajoutée la distance s_{min} dans la direction D. Si une frontière séparant deux volumes géométriques est traversée sur le trajet menant à la nouvelle position alors annuler l'interaction, déplacer la particule sur la frontière (volume suivant) et recommencer l'algorithme depuis le début.
7. Utiliser un "modèle" pour déterminer les effets de l'interaction telle qu'une perte d'énergie impliquant une nouvelle énergie E_n pour la particule suivie, la création d'éventuelles particules secondaires et, enfin, la nouvelle direction de la particule D_n .
8. Mettre à jour D, P et E :

$$\begin{cases} D = D_n \\ P = P_n \\ E = E_n \end{cases}$$

9. Si la particule suivie est toujours "valide", c'est à dire que son énergie n'est pas inférieure à un seuil prédéterminé et qu'elle se trouve toujours dans le volume simulé, alors retourner au début de l'algorithme avec les nouvelles valeurs de D, P et E. A l'inverse, si la particule suivie n'est plus "valide", alors la retirer de la simulation et recommencer l'algorithme avec les éventuelles particules secondaires générées durant le transport de la particule primaire.

ALGORITHME 1.1 – Version simplifiée du transport de particule avec la méthode de Monte Carlo.

tenir le code. Cette collaboration a permis d'introduire de nombreux modèles physiques allant de l'électromagnétisme à la physique hadronique rendant ainsi le code extrêmement flexible et polyvalent. Geant4 utilise à la fois une approche d'histoire condensée et une approche "pas à pas". Il s'agit donc d'un code de structure de trace capable d'utiliser les histoires condensées. Le choix de la méthode à mettre en œuvre pour un calcul donné appartient à l'utilisateur qui peut également décider de combiner les deux principes dans une même simulation. Les aspects physiques du fonctionnement de Geant4 sont détaillés dans le manuel de référence sur la physique [IVANCHENKO et al., 2014] tandis que les aspects informatiques sont abordés dans la documentation associée au code.

Le transport des particules dans Geant4 fait appel à des notions spécifiques étroitement liées au déroulement d'une simulation Geant4 typique. Ainsi, dans Geant4, tout commence avec le **run**. Le run initialise, déclenche et termine une simulation. Dans un run, des particules dites "primaires" sont introduites dans la simulation et transportées. L'introduction d'une nouvelle particule primaire correspond à un **event** (événement). Un event débute avec la création de la nouvelle particule et se termine lorsque cette dernière ainsi que toutes les éventuelles particules secondaires qu'elle a pu mettre en mouvement ont été transportées. Durant chaque event, plusieurs **tracks** (traces) sont simulés. Un track représente une particule et son état à un instant donné dans la simulation. Le track contient donc les informations relatives au type, à l'énergie, à la position et à la direction d'une particule. L'évolution d'un track entre la création d'une particule et son retrait de la simulation constitue l'histoire de cette particule. Ainsi, l'introduction d'une particule primaire génère au minimum un track mais si cette particule interagit avec le milieu ambiant et génère des particules secondaires alors de nouveaux tracks seront créés et placés dans une pile appelé **stack**. Lorsque Geant4 a complété le transport d'un track donné, il choisit alors le track suivant en sélectionnant celui qui se trouve tout en haut du stack (de la pile). Chaque track est défini à partir de **steps** (pas) qui représentent les étapes du transport d'une particule. Chaque step est divisé en trois sous parties (cf. figure 1.8) : **pre-step**, **along-step** et **post-step**. Le pre-step est l'état initial de la particule avant que le transport soit effectué tandis que le post-step correspond à son état final. Entre ces deux états, la particule est transportée et ses caractéristiques sont éventuellement ajustées si une interaction se produit. Along-step est utilisé dans le cadre d'une simulation avec des histoires condensées car il permet de répartir le long du trajet effectué un dépôt d'énergie. Il est intéressant de noter que les dépôts d'énergie sont localisés en post-step dans le cas d'une simulation "pas à pas". Pour plus de clarté, les schémas de la figure 1.7 montrent l'articulation de ces différents concepts lors d'une simulation.

La mise en œuvre concrète de ces concepts est effectuée dans le code source de Geant4. Ce dernier est divisé en plusieurs modules, également nommés "packages", gérés par les différents groupes de la collaboration Geant4. Les principaux modules du code sont présentés en figure 1.9 et listés ci-dessous :

- Global** Système d'unités, constantes, tirages aléatoires et divers aspects mathématiques.
- Materials** Propriétés physiques des matériaux.
- Particles** Propriétés physiques des particules.
- Geometry** Description d'une structure géométrique dans laquelle suivre les particules.
- Track** Définit les concepts de "track" et "step". Un track est l'histoire d'une particule de sa création à sa suppression de la simulation tandis qu'un step un "pas" effectué durant cette histoire.
- Processes** Contient les modèles en charge des interactions physiques de type électromagnétique (leptons, photons, hadrons et ions) et hadronique.

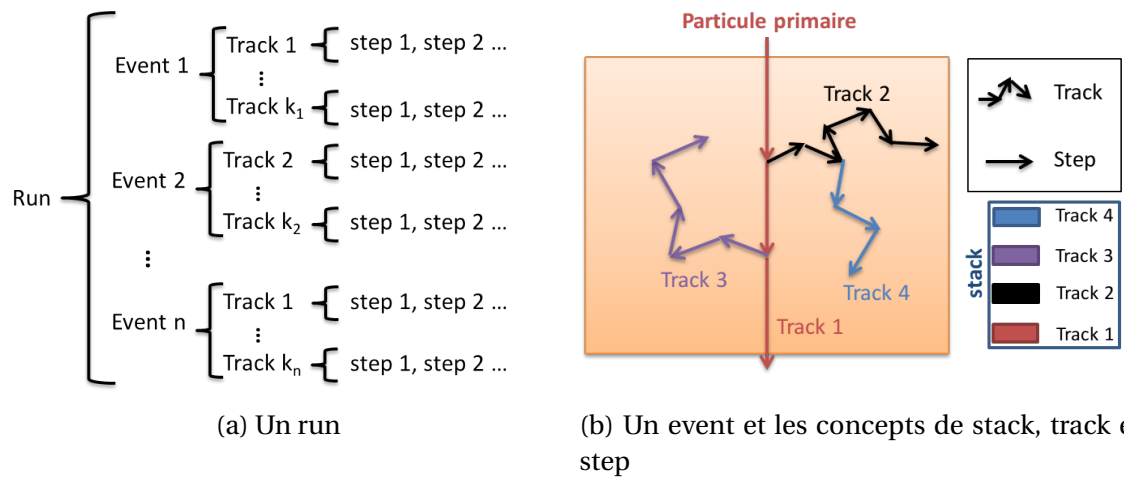


FIGURE 1.7 – Articulation des concepts de run, event, track et step lors d'une simulation Geant4.

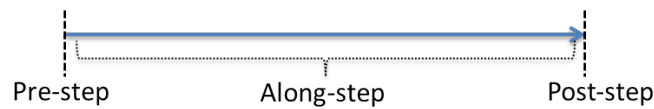


FIGURE 1.8 – Les trois actions effectuées lors d'un step (flèche bleue).

Tracking Gère l'évolution d'une particule (track) au fur et à mesure que son histoire est simulée.

Event Prend en charge et organise l'ensemble des événements déclenchés par le lancement d'une particule dans la simulation (les tracks).

Run Débute, fait tourner et termine la simulation. Ce module gère l'ensemble de la simulation, c'est à dire les particules à prendre en compte, leurs histoires et la géométrie dans laquelle les suivre.

Readout Permet d'accumuler des informations dans certains volumes géométriques.

Autres **Visualization**, **Persistency** et **Interfaces** sont des modules transverses faisant appel à l'ensemble des modules précédemment cités. Ils mettent à disposition d'une application Geant4 des capacités de visualisation, de sauvegarde et un système de macro pour interagir avec la simulation.

Une particularité notable de Geant4 se situe au niveau de son module Geometry qui facilite grandement la définition de géométries pour le transport de particules. En effet, ce module permet à un utilisateur de définir une géométrie dans laquelle transporter les particules à partir de plusieurs volumes de bases implémentés par défaut : cube, cylindre, ellipsoïde, sphère etc. Chacun de ces volumes possède un ensemble de paramètres ajustables par l'utilisateur et, si ces volumes devaient s'avérer insuffisants, il est également possible de les combiner au moyen d'opérations booléennes de type union et soustraction. La construction de tels volumes géométriques se traduit par la création d'objets Geant4 appelés **volumes solides**. Un volume solide contient l'ensemble des informations concernant la forme en 3D du volume créé par l'utilisateur. Cependant, dans le cadre du transport de particules, la définition d'une forme 3D n'est pas suffisante. En effet, il faut également spécifier le matériau correspondant ainsi que la masse volumique du volume créé. Ces spécifications se font au travers d'un autre objet Geant4 : le **volume logique**. Enfin, la dernière étape requise pour que Geant4 transporte des particules dans une géométrie, est le placement des volumes créés par l'utilisateur. Il s'agit de définir un

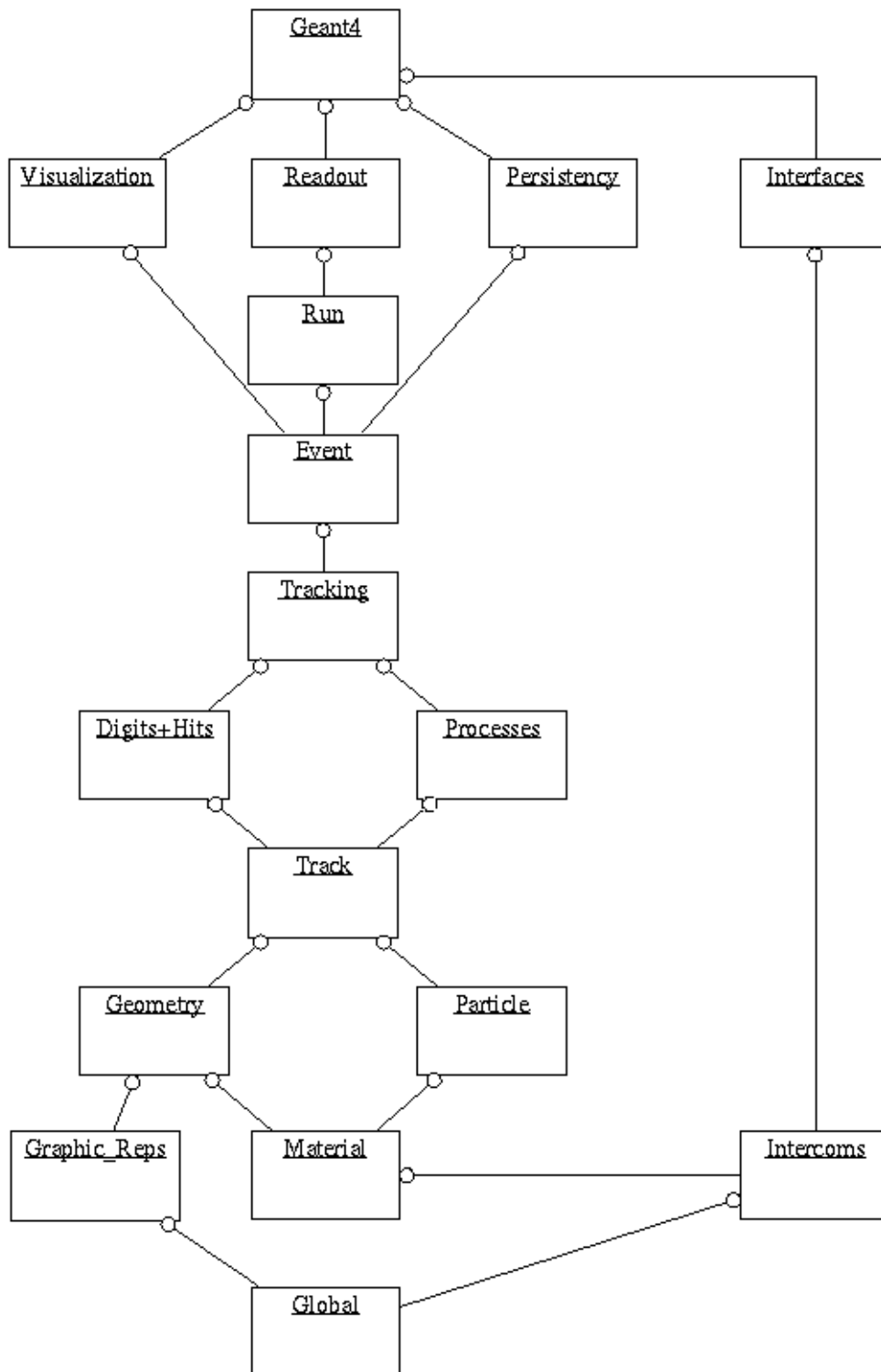


FIGURE 1.9 – Les principaux modules de Geant4 en anglais (source : geant4.org). Dans ce graphique, les modules les plus bas sont utilisés par les modules qui les surplombent. Les modules situés en bas du graphique sont donc les modules de base sur lesquels s'appuient les autres composants.

objet Geant4 **placement** qui contiendra les informations relatives à la position et l'orientation du volume. Comme plusieurs "placements" peuvent être effectués pour un même ensemble "volume logique/volume solide", cela permet de positionner plusieurs fois un volume sans avoir à dupliquer les informations relative à son matériau ou à sa forme 3D en mémoire. Pour résumer, lors de la définition d'un volume géométrique, un utilisateur créé tout d'abord un volume solide avec lequel il définit la forme 3D du volume. Ensuite, l'utilisateur associe ce volume solide à un volume logique spécifiant le matériau constitutif du volume. Enfin, l'utilisateur positionne et oriente le volume nouvellement créé dans la simulation au moyen d'un "placement". Le placement d'un volume s'effectue de manière hiérarchique et relative. Cela signifie que tout placement est fait relativement à un volume parent préalablement positionné. A cet effet, un volume racine est automatiquement créé lors de l'initialisation d'une simulation : le world.

1.2.3 L'extension Geant4-DNA

Geant4 permet de simuler de nombreux processus physiques parmi lesquels se trouvent les processus du groupe électromagnétique. Ces derniers sont particulièrement intéressants dans le cadre de ce travail car certains d'entre eux appartiennent à la catégorie "Geant4-DNA" qui se spécialise dans le transport de particules de faibles énergies (de quelques eV à quelques MeV). Le projet Geant4-DNA [BERNAL et al., 2015; INCERTI et al., 2010] fut initié en 2001 dans le but de simuler les effets biologiques précoces de différent types de particules à des échelles cellulaire et subcellulaire. Dans ce cadre, le projet se focalise particulièrement sur les effets liés à l'ADN et s'articule autour de trois axes de développement principaux :

1. Simulation de l'étape physique : développement de modèles physiques pour effectuer le transport de particules dans la matière biologique.
2. Simulation des étapes physico-chimique et chimique : mise en place de modèles capables de simuler les processus chimiques de la radiolyse de l'eau (détaillés dans le chapitre 1.3).
3. Introduction de modèles biologiques précoces par l'intermédiaire de géométries.

D'autres projets de recherches et codes de calculs, parfois antérieurs à Geant4-DNA, partagent ces objectifs et mettent en œuvre des approches très similaires. Le code PARTRAC [FRIEDLAND et al., 2011] écrit en FORTRAN permet notamment d'aller très loin dans la simulation en intégrant des processus de réparation des dommages à l'ADN. Les codes KURBUC [NIKJOO et GIRARD, 2012] et PHITS [WILSON et al., 2004] sont similaires et permettent la simulation des étapes physique, physico-chimique et chimique. L'ensemble de ces codes, Geant4-DNA compris, approxime le milieu biologique dans lequel les particules sont transportées à de l'eau liquide. L'influence des constituants biologiques étant généralement considérés comme négligeable au regard de celle de l'eau liquide qui est majoritairement présente dans les cellules humaines. Le code CPA-100 est une exception à cet état de fait car il intègre des sections efficaces de l'ADN calculées à partir de celles de l'eau liquide [TERRISSOL et al., 2002].

Les processus physiques implémentés dans Geant4-DNA sont, pour la plupart, négligés dans les codes de Monte Carlo classiques en raison des faibles dépôts d'énergie qui leur sont associés. Cependant, l'étude des interactions particule/matière à une échelle subcellulaire implique de considérer des processus physiques avec des libres parcours moyens de petites tailles. La grande majorité des interactions suscitées par ces processus impliquent des électrons dits "secondaires" car mis en mouvement à partir du transport d'une particule primaire dans la matière. Dans Geant4-DNA, les processus physiques

liés à ces électrons sont décrits de manière à les transporter jusqu'à leur thermalisation (quelques eV) et les trajectoires générées sont intégralement simulées grâce à l'approche "pas à pas". Les processus physiques impliqués, et les modèles qui leur sont associés dans le code, sont abondamment décrits dans la littérature [BERNAL et al., 2015; DOS SANTOS, 2013; FRANCIS, 2007; INCERTI et al., 2010; KARAMITROS, 2012; VILLAGRASA et al., 2010]. Dans Geant4-DNA (Geant4 version 10.1), les processus élastiques ne concernent que les électrons et les trajectoires des autres particules (protons, photons, ions) sont considérées comme rectilignes. Les processus inélastiques décrivent la manière dont les particules transportées perdent leur énergie dans la matière. Le tableau 1.2 liste les différents modèles physiques inclus dans Geant4-DNA ainsi que leurs limites en énergie. Ces modèles peuvent être issus de calculs théoriques, d'analyses semi-empiriques ou encore de données expérimentales.

Le code de Geant4-DNA est articulé autour des concepts de processus et modèle et divisé en cinq grands ensembles :

- **Management** contient une partie de l'infrastructure requise par Geant4-DNA (informations relatives à la mémoire utilisée, liste de tracks ...).
- **Processes** définit les processus utilisés par Geant4-DNA (elastic, ionisation, excitation etc.)
- **Models** contient les modèles physiques de Geant4-DNA.
- **Molécules** sert à la simulation de la radiolyse de l'eau (cf. chapitre 1.3).
- **Utils** inclut différents éléments généraux.

A l'intérieur de ces ensembles se trouve également le code du module chimie de Geant4-DNA (cf. chapitre 1.3). D'un point de vue général, à chaque processus Geant4-DNA est lié à un ou plusieurs modèles qui le caractérisent. Ces processus sont également enregistrés au sein de Geant4 de manière à être sollicités lors d'une simulation. La sollicitation d'un processus Geant4-DNA par Geant4 équivaut à la demande d'un libre parcours moyen correspondant au processus concerné. Ce libre parcours moyen est calculé par Geant4-DNA à l'aide des modèles associés au processus.

TABLEAU 1.2 – Modèles physiques de Geant4-DNA et leurs domaines de validité en énergie (Geant4.10.1).

Particules	e ⁻	P	H	He, He ⁺ , He ²⁺	C, N, O, Fe ...
Diffusion élastique	9 eV - 1 MeV Screened Rutherford 7 eV - 1 MeV Champion	-	-	-	-
Excitation	9 eV - 1 MeV Born	10 eV - 500 keV Miller Green 500 keV- 100 MeV Born	10 eV - 500 keV Miller Green	Ajustée à partir des modèles du proton 1 keV - 400 MeV	-
Échange de charge	-	100 eV - 100 MeV Dingfelder	100 eV - 100 MeV Dingfelder		-
Ionisation	11 eV - 1 MeV Born	100 eV - 500 keV Rudd 500 keV - 100 MeV Born	100 eV - 100 MeV Rudd		Ajusté à partir du modèle du proton 0.5 MeV/u - 10 ⁶ MeV/u
Excitation vibrationnelle	2 eV - 100 eV Michaud et al.	-			
Attachement dissociatif	4 eV - 13 eV Melton				

1.2.4 Utiliser Geant4/Geant4-DNA

Geant4 étant conçu comme une boîte à outils, un utilisateur Geant4 se doit d'écrire lui-même un programme en C++ qui utilise les outils mis à sa disposition. Ce programme appelé "application utilisateur" (AU) doit respecter une structure standard mais, néanmoins, adaptable de manière à satisfaire les besoins spécifiques de nombreux utilisateurs. Ainsi, une application utilisateur Geant4 est constituée d'un socle général auquel l'utilisateur peut ajouter des éléments pour la personnaliser et récupérer des informations durant la simulation. Le socle est constitué des éléments suivants :

- Un fichier *CMakeLists.txt* contenant divers réglages permettant de lier le code de l'application utilisateur à Geant4.
- Le main (*main.cc*) qui est le cœur de l'application utilisateur. C'est dans ce fichier que se déclenche l'initialisation de la simulation, le lancement des particules primaires et le nettoyage de la mémoire à la fin de l'exécution. Le main est également l'endroit où les composants de l'application utilisateur sont mis en relation.
- Le DetectorConstruction (*DetectorConstruction.cc* et *.hh*) contient la définition de la géométrie associée à la simulation.
- La PhysicsList (*PhysicsList.cc* et *.hh*) permet de déterminer les processus et sélectionner les modèles physiques à utiliser durant la simulation.
- Le PrimaryGenerator (*PrimaryGenerator.cc* et *.hh*) est utilisé pour définir les caractéristiques des particules primaires de la simulation ainsi que leurs positions de départ.
- Le ActionInitialization (*ActionInitialisation.cc* et *.hh*) lie au reste du programme les classes optionnelles éventuellement créées par l'utilisateur et prend en charge une possible parallélisation de la simulation.

Tandis que les éléments optionnels les plus couramment utilisés sont :

- Le RunAction (*RunAction.cc* et *.hh*) contient deux méthodes appelées respectivement au début et à la fin de la simulation : *BeginOfRunAction* et *EndOfRunAction*. C'est généralement au travers de ces méthodes qu'est défini le fichier qui recevra les données générées par la simulation.
- Le EventAction (*EventAction.cc* et *.hh*) et ses méthodes *BeginOfEvent* et *EndOfEvent* appelées à chaque début et fin d'événement.
- Le SteppingAction (*SteppingAction.cc* et *.hh*) et les méthodes *PreStepUserAction* et *PostStepUserAction* appelées respectivement en début et fin de chaque step. Ces deux méthodes sont utilisées pour collecter les données à enregistrer dans le fichier créé dans le RunAction.

Au début d'une nouvelle simulation, l'ensemble des éléments de l'AU sont utilisés conjointement avec les outils fournis par Geant4-DNA. L'enchaînement ainsi créé est illustré en figure 1.10 où chaque étape est numérotée. Directement après le lancement d'une simulation, l'étape 1 est enclenchée pour initialiser à la fois les éléments de l'application utilisateur et les outils fournis par Geant4. En étape 2, l'AU déclenche le tirage d'un nombre n_p de particules dans la simulation. Ces particules seront générées selon les caractéristiques spécifiées dans le PrimaryGenerator (type, énergie, position, direction...). Lors des étapes 3, 4 et 5, Geant4 se prépare à introduire de nouvelles particules dans la simulation et débute un nouveau run. A cette occasion, Geant4 fait appel à une méthode personnalisable par l'utilisateur intitulée *BeginOfRunAction* (étape 4). Les particules primaires sont introduites séquentiellement dans la simulation au travers d'événements (étape 6). A chaque

particule correspond un nouvel event dont la création donne lieu à l'appel de la méthode personnalisable *BeginOfEventAction* (étape 7). Chaque nouvelle particule est ensuite transportée dans la simulation au sein d'un track (étape 8). Les interactions d'une particule primaire pouvant générer des particules secondaires, chaque event donne lieu à plusieurs tracks. La création d'un nouveau track (étape 8) déclenche l'appel de la méthode personnalisable *PreUserTrackingAction* (étape 9) et sa destruction l'appel à *PostUserTrackingAction* (étape 13). Chaque nouveau track représente une particule à transporter jusqu'à son retrait de la simulation, (= énergie inférieure à un seuil prédéterminé ou sortie du world). Une fois la particule retirée de la simulation, le track est détruit (étape 12). Le transport d'une particule s'effectue au moyen d'une série de steps (étape 10). Le calcul de chaque step donne lieu à l'appel de la méthode personnalisable *UserSteppingAction* (étape 11). Lorsque tous les tracks générées lors d'un événement sont détruits alors l'event est clôturé (étape 14) et la méthode personnalisable *EndOfEventAction* appelée (étape 15). Une fois tous les events simulés, le run se termine (étape 16) et appelle la méthodes personnalisable *EndOfRunAction* (étape 17). Enfin, la mémoire utilisée lors de la simulation est libérée (étape 18) et le programme terminé.

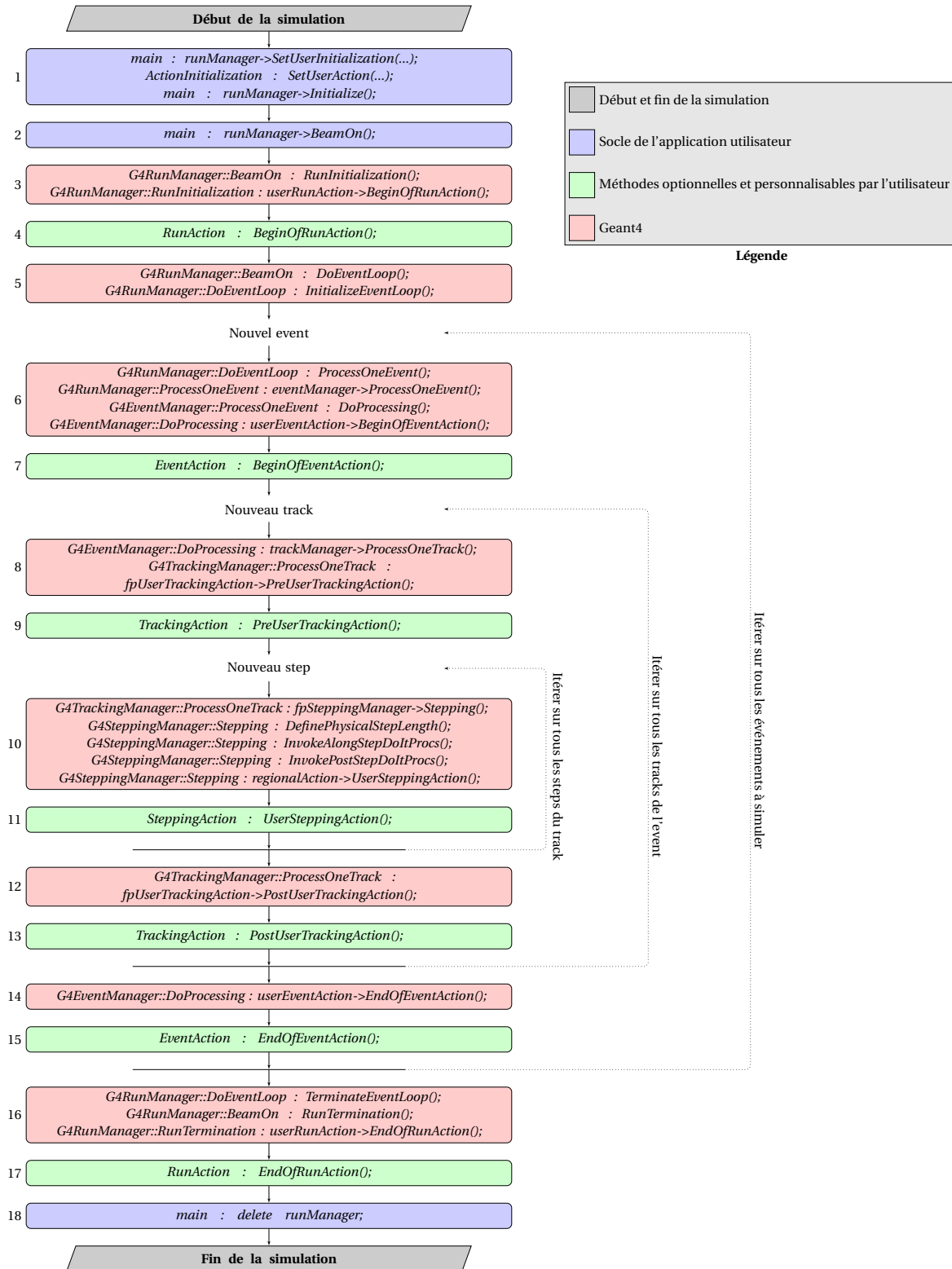


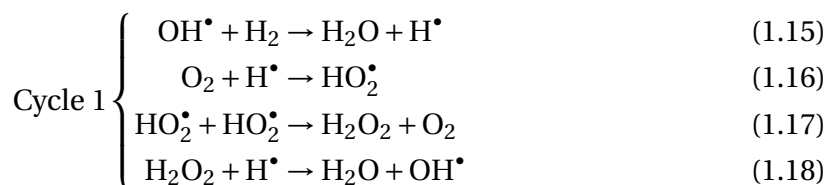
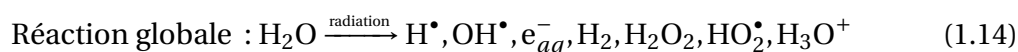
FIGURE 1.10 – Logigramme simplifié des méthodes utilisées lors d'une simulation Geant4 (mode séquentiel).

1.3 Simulation de la radiolyse de l'eau

1.3.1 Généralités

La radiolyse de l'eau liquide est un processus induisant la dissociation de molécules d'eau suite au passage de particules ionisantes. Elle est abondamment utilisée dans le cadre de la chimie sous rayonnement où des rayonnements ionisants sont utilisés comme déclencheurs de réactions chimiques. La naissance de cette discipline remonte à la fin du 19^{ème} siècle [KERNBAUM, 1910] et c'est en 1914 que le chimiste André Debierne montre que la décomposition de l'eau exposée à des radiations engendre des espèces chimiques, appelées **produits radiolytiques**, parmi lesquelles se trouvent des **radicaux libres**. Les radicaux sont des espèces chimiques particulièrement réactives en raison de leurs cortèges électroniques qui possèdent un ou plusieurs électrons de valence non appariés. La nature radicalaire d'une espèce chimique est spécifiée à l'écrit par un point placé juste après son nom. La radiolyse de l'eau n'est qu'un moyen de production des radicaux parmi d'autres. En effet, les radicaux sont également produits naturellement dans l'organisme humain par la respiration mitochondriale, certains organites (peroxysomes), diverses réactions d'oxydoréduction, la phagocytose etc.

Historiquement, les chimistes travaillant sur la radiolyse de l'eau ont remarqué que l'ajout d' H_2 préalablement à une irradiation avec des particules avec de faibles TEL ne permettait plus d'observer la décomposition de l'eau liquide. A l'inverse, l'ajout d' O_2 entraînait systématiquement une décomposition observable quelque soit le TEL des particules employées. Ces fluctuations dans les observations expérimentales s'expliquent par la prise en compte d'une série de réactions chimiques se déclenchant dès l'apparition des premiers produits radiolytiques [ALLEN, 1961; BALDACCHINO et HICKEL, 2008] :



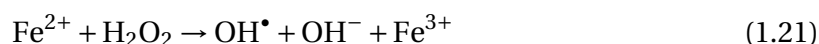
La réaction 1.14 représente la radiolyse de l'eau de manière globale et l'ensemble des produits radiolytiques générés. Suite à l'apparition de ces espèces chimiques dans le milieu, un premier ensemble de réactions débute avec conversion d'un radical OH^\bullet en radical H^\bullet (cf. réaction 1.15). Ce dernier est particulièrement réactif et réduit donc très rapidement l' O_2 en H_2O_2 via les réactions 1.16 et 1.17. A ce stade, la réaction 1.18 permet d'amorcer un cycle (cf. Cycle 1) par la production d' OH^\bullet à partir de H^\bullet et d' H_2O_2 . Cet OH^\bullet pouvant ensuite entraîner une nouvelle réaction 1.15 et débiter un nouvel enchaînement. Si ce cycle se déroule sans accrocs alors les radicaux sont peu à peu absorbés. Cependant, il est nécessaire pour cela que la concentration d' O_2 soit faible. En effet, une présence significative d' O_2 entraînerait quasi-systématiquement l'apparition de la réaction 1.16 et, ainsi, la disparition de tous les H^\bullet avant qu'ait pu avoir lieu une réaction 1.18. Le cycle serait alors brisé et les réactions encore actives généreraient de nombreux HO_2^\bullet ensuite observés par les chimistes comme marqueurs de la décomposition de l'eau liquide. La réaction 1.19 représente une alternative au cycle précédemment présenté. En effet, le H_2O_2 généré

par la réaction 1.17 peut provoquer la réaction 1.18 ou la réaction 1.19. Si une trop grande quantité de H_2O_2 est présente alors les réactions 1.19 et 1.17 se succèdent et produisent de l' O_2 . Comme précédemment, l'augmentation de la quantité d' O_2 arrête le cycle et provoque l'augmentation de la quantité d' HO_2^\bullet . Les réactions 1.19 et 1.15 étant également en compétition, la présence de H_2 est nécessaire pour s'assurer que tous les OH^\bullet ne soient pas consommés par la réaction 1.19 au détriment de la réaction 1.15. Finalement, l'ensemble de ces réactions effectue la transformation représentée en réaction 1.20.



Cette dernière réaction explicite pourquoi l'ajout d' H_2 et le TEL ont une influence sur la décomposition de l'eau liquide. En effet, à bas TEL, les quantités d' O_2 , H_2 et H^\bullet sont équilibrées et le cycle de réactions peut avoir lieu reproduisant ainsi la réaction globale 1.20. En revanche, à haut TEL, il y a trop de O_2 et H_2 (car ajouté) par rapport à la quantité d' H^\bullet ce qui provoque l'arrêt du cycle [BALDACCHINO et HICKEL, 2008]. Cet arrêt rend la décomposition de l'eau visible pour un expérimentateur expliquant ainsi les observations précédemment mentionnées. Ces enchaînements de réactions montrent bien que les processus chimiques mis en oeuvre lors de la radiolyse de l'eau sont complexes. D'autant plus que les concentrations des différentes espèces chimiques impliquées peuvent varier fortement lors de l'introduction d'éléments extérieurs, redéfinissant par là même les équilibres mentionnés précédemment. C'est cette complexité qui a poussé les chimistes à utiliser des moyens numériques pour simuler l'évolution de leurs systèmes en fonction de différents paramètres.

Le radical OH^\bullet est particulièrement intéressant dans ce travail de thèse car il peut être à l'origine de dommages à l'ADN difficilement réparables par la cellule touchée. Ce radical peut être créé à partir de la radiolyse de l'eau telle que présenté ci-dessus mais également à partir de la réaction dite de "Fenton" :



Au final, la génération de radicaux OH^\bullet dans une cellule peut provenir d'un processus combinant radiolyse de l'eau, réaction Fenton et encore d'autres éléments. Dans le cadre de la réaction Fenton, l'ion Fe^{3+} peut être converti à nouveau en Fe^{2+} et former ainsi un cycle dit **cycle de Haber-Weiss**. Dans un contexte biologique favorable, ce cycle peut entraîner une création massive de radicaux OH^\bullet dans et autour d'une cellule et donc accroître significativement la probabilité d'occurrence de dommages à l'ADN.

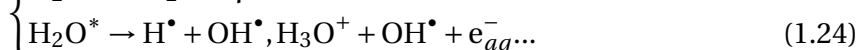
1.3.2 La chimie dans Geant4-DNA

Depuis 2014, le code Geant4-DNA a été étendu pour rendre possible la simulation des étapes physique, physico-chimique et chimique au sein d'un unique outil. Cette extension, nommée "module chimie" dans ce travail, consiste en une nouvelle infrastructure (Interactive Tracking, IT) associée à de nouveaux processus capables de simuler la radiolyse de l'eau. La théorie mise en oeuvre dans ce module chimie est déjà décrite exhaustivement dans la littérature [KARAMITROS, 2012; KARAMITROS et al., 2014, 2011] et le lecteur est encouragé à s'y référer pour obtenir des informations complémentaires.

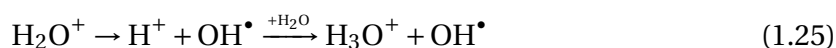
Les molécules d'eau instables

Les dépôts d'énergie occasionnés par le passage d'une particule dans l'eau liquide peuvent provoquer l'instabilité de molécules d'eau en les ionisant ou en les excitant. Ces

molécules peuvent ensuite se dissocier pour retourner dans un état stable. Il s'agit de l'étape physico-chimique qui a lieu d'environ 10^{-14} à 10^{-12} seconde après le début de l'irradiation. Une molécule d'eau ionisée peut suivre deux voies pour se stabiliser : la recombinaison ou la dissociation. La recombinaison (cf. réaction 1.22), généralement nommée "recombinaison géminée dissociative", consiste à absorber un électron pour combler le vide occasionné par l'ionisation. Suite à cette absorption, la molécule est dans un état d'excitation également instable. Le retour vers la stabilité peut alors s'effectuer avec l'émission d'un photon (cf. réaction 1.23) ou via la dissociation de la molécule (cf. réaction 1.24). Dans Geant4-DNA, cette dernière peut avoir lieu selon les différents canaux présentés dans le tableau 1.3.



Si la molécule ionisée se stabilise non pas au travers d'une recombinaison mais directement avec une dissociation alors une seule possibilité de dissociation est prise en compte :



Enfin, une molécule d'eau directement excitée par le passage d'une particule physique se comportera comme une molécule excitée suite à une ionisation et une recombinaison. C'est à dire qu'elle empruntera l'un des canaux spécifié dans le tableau 1.3.

TABLEAU 1.3 – Les canaux de dissociation présents dans Geant4-DNA pour une molécule d'eau liquide excitée [KARAMITROS, 2012].

Type d'excitation (état électronique)	Canaux de dissociation	Probabilité
A^1B_1	$\text{OH}^\bullet + \text{H}^\bullet$	65%
	$\text{H}_2\text{O} + \gamma$	35%
B^1A_1	$\text{OH}^\bullet + \text{H}_3\text{O}^+ + \text{e}_{aq}^-$	55%
	$\text{H}_2 + 2\text{OH}^\bullet$	15%
	$\text{H}_2\text{O} + \gamma$	30%
Rydberg	$\text{OH}^\bullet + \text{H}_3\text{O}^+ + \text{e}_{aq}^-$	50%
	$\text{H}_2\text{O} + \gamma$	50%

La traversée d'un milieu aqueux par une particule génère de nombreuses interactions à l'origine de la perte d'énergie de la particule incidente. Dans de nombreux cas, cette particule se trouve être un électron (particule secondaire) et, lorsque son énergie cinétique atteint quelques meV, ce dernier atteint l'équilibre thermique. A ce moment, il peut être "capturé" par les molécules d'eau qui l'entourent pour former une nouvelle espèce chimique : l'**électron solvaté** noté e_{aq}^- [HART et BOAG, 1962]. Cette capture, illustrée en figure 1.11, est causée par l'interaction entre la charge négative de l'électron et les moments dipolaires électriques des molécules d'eau l'entourant. Il est intéressant de noter que l'électron solvaté est le produit radiolytique le plus présent quantitativement lors de la radiolyse de l'eau [KARAMITROS, 2012].

Parallèlement à ce processus de solvatation, il est également possible qu'un électron de faible énergie soit capturé par une unique molécule d'eau. Cette dernière devient alors instable et se dissocie selon la réaction 1.26. Il s'agit de l'attachement dissociatif [MELTON,

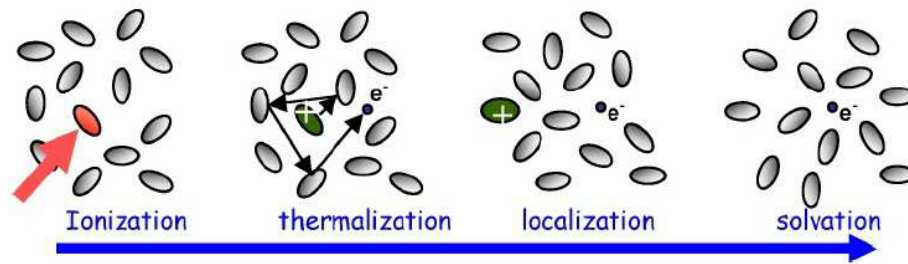
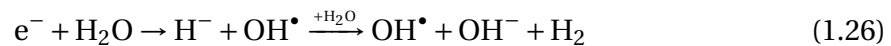


FIGURE 1.11 – Schéma représentant le processus de création d'un électron solvaté [SPOTHEIM-MAURIZOT et al., 2008].

1972]. Dans Geant4, la capture d'un électron par une molécule d'eau est prise en charge lors de l'étape physique avec un processus et un modèle dédiés tandis que la dissociation de la molécule d'eau est gérée par le module "chimie".



Diffusion et réactions des espèces chimiques

Entre environ 10^{-10} et 10^{-6} seconde après le début de l'irradiation, les espèces chimiques créées lors de l'étape physico-chimique vont se déplacer aléatoirement dans le milieu et, parfois, se rencontrer et réagir entre elles. La répartition spatiale initiale de ces espèces chimiques dépend fortement du rayonnement à l'origine des perturbations dans le milieu et notamment de son TEL. Pour le cas des ions, des particules de faibles TEL donneront lieu à des distributions initiales en forme de grappes tandis que des particules avec de forts TEL induiront des distributions plus homogènes le long de leur parcours. Précisons également que lors des premiers instants de diffusion les espèces sont proches avec une probabilité significative de se rencontrer, ce qui est de moins en moins vrai au fur et à mesure qu'elles se dispersent dans le milieu.

La chimie dans Geant4-DNA s'appuie sur une représentation du type particule-continuum non compartimentée où les molécules sont des "particules" évoluant dans un "continuum" constitué d'eau liquide. Cette représentation est associée à un traitement stochastique de la diffusion des molécules s'appuyant sur la théorie de Smoluchowski et l'équation d'Ermak-McCammon. Ce choix est adapté aux systèmes dont les réactifs sont répartis de manière hétérogène et dont le temps simulé ne dépasse pas l'ordre de la microseconde. Des hypothèses simplificatrices sont effectuées d'emblée dans le code :

- Le milieu est isotrope et constitué d'eau liquide.
- Chaque paire de molécule diffuse indépendamment des autres molécules.
- Les molécules sont sphériques.

Ces simplifications permettent la mise en place d'une simulation où les molécules, qui représentent les espèces chimiques générées lors de l'étape physico-chimique, diffusent et réagissent dans un solvant isotrope (eau liquide). Cette diffusion est simulée au moyen d'une succession de pas en temps durant lesquels les molécules sont déplacées simultanément. Suite à chacun de ces pas en temps, les positions des molécules sont passées en revue et si les sphères qui les représentent sont à leur distance de réaction alors une réaction a lieu. Cette dernière consomme les deux molécules "réactifs" et crée des "produits". La distance que peut parcourir une molécule lors d'un pas en temps dt est calculée à

partir de l'équation d'Ermak-McCammon [ERMAK et McCAMMON, 1978] :

$$x(t_0 + \tau) = x_0 + \frac{F(x_0)}{m\gamma} \Big|_{t_0} \cdot \tau + \sqrt{2D\tau} \cdot N(0, 1) + O(\tau^2) \quad (1.27)$$

avec t_0 le temps initial, τ le pas en temps, x_0 la position initiale selon x , $F(x_0)$ une force extérieure appliquée à la molécule, m la masse de la molécule, $\gamma = \alpha/m$ le coefficient de friction, α ($[masse][temps]^{-1}$) une constante liée à la viscosité, D le coefficient de diffusion et $N(0, 1)$ un nombre aléatoire tiré à partir d'une gaussienne de moyenne 0 et de variance 1. Cette équation permet de reproduire le caractère Brownien du déplacement des molécules. Elle peut être simplifiée car aucune force extérieure $F(x_0)$ n'est appliquée aux molécules dans Geant4-DNA, ce qui implique :

$$\frac{F(x_0)}{m\gamma} \Big|_{t_0} = 0$$

L'équation 1.27 devient donc :

$$x(t_0 + \tau) = x_0 + \sqrt{2D\tau} \cdot N(0, 1) + O(\tau^2) \quad (1.28)$$

En remplaçant τ par dt tout en appliquant l'équation 1.28 aux trois dimensions (x, y, z), nous pouvons écrire :

$$\left\{ \begin{array}{l} dx = \sqrt{2D \cdot dt} \cdot N_x(0, 1) \\ dy = \sqrt{2D \cdot dt} \cdot N_y(0, 1) \\ dz = \sqrt{2D \cdot dt} \cdot N_z(0, 1) \end{array} \right\} \Rightarrow \vec{r}_{t+dt} = \vec{r}_t + \begin{pmatrix} dx \\ dy \\ dz \end{pmatrix} \quad (1.29)$$

où \vec{r}_t est la position d'une molécule à l'instant t et \vec{r}_{t+dt} sa positions à l'instant $t + dt$. Ainsi, la position d'une molécule, placée en \vec{r}_t à l'instant t , après un pas en temps dt est calculée à partir de l'équation 1.29 et du tirage de trois variables aléatoires $N_x(0, 1)$, $N_y(0, 1)$ et $N_z(0, 1)$. Cependant, l'usage de cette équation nécessite deux éléments :

- La valeur du coefficient de diffusion D de la molécule considérée. Dans le code, cette valeur est fournie pour chaque molécule en tant que donnée d'entrée. Les coefficients de diffusion utilisés dans Geant4-DNA sont listés dans le tableau 1.4.
- La durée dt du pas en temps simulé. Cette durée est calculée préalablement à l'exécution de chaque pas en temps.

TABEAU 1.4 – Les coefficients de diffusion des espèces chimiques pris en compte dans Geant4-DNA [KARAMITROS et al., 2011].

Espèce chimique	Coefficient de diffusion ($10^{-9} \text{m}^2 \text{s}^{-1}$)
e_{aq}^-	4.9
OH^\bullet	2.8
H^\bullet	7.0
H_3O^+	9.0
H_2	4.8
OH^-	5.0
H_2O_2	2.3

Le calcul de la durée dt d'un pas en temps est effectué à l'aide de la théorie de Smoluchowski. Cette théorie est théoriquement compatible avec un transport des molécules

basé sur l'équation d'Ermak-McCammon (cf. équation 1.27) et, dans le cas de Geant4-DNA, cela a été vérifié en pratique [KARAMITROS et al., 2014]. La théorie de Smoluchowski est construite à partir de l'équation du même nom présentée ci-dessous (1D) :

$$\frac{\partial p(x, t)}{\partial t} = -\frac{\partial}{\partial x} \left(\frac{F}{m\gamma} \cdot p(x, t) \right) + \frac{\partial^2}{\partial x^2} (D \cdot p(x, t)) \quad (1.30)$$

avec $p(x, t)$ la densité de probabilité d'une molécule au temps t et à la distance x de l'origine, F une force extérieure, m la masse de la molécule considérée, γ la constante de friction et D le coefficient de diffusion. Cette équation n'est valable que dans le cas où la constante de friction γ est très large vis à vis de l'accélération a :

$$\gamma \cdot v \gg a$$

où v est la vitesse. L'équation 1.30 est généralisable en trois dimensions :

$$\frac{\partial p(\vec{r}, t)}{\partial t} = \vec{\nabla} \cdot \left(D \cdot \left(\vec{\nabla} p(\vec{r}, t) - \beta F(\vec{r}) \cdot p(\vec{r}, t) \right) \right) \quad (1.31)$$

avec $\beta = \frac{1}{k_B T}$, k_B la constante de Boltzmann et T la température. La solution de cette généralisation étant donnée par :

$$p(\vec{r}, t) = \frac{4\pi r^2}{(4\pi D t)^{3/2}} \cdot e^{-\frac{r^2}{4Dt}} \quad (1.32)$$

Le principe est d'utiliser la théorie Smoluchowski pour calculer un pas en temps dt qui corresponde au temps minimal nécessaire pour que les deux molécules A et B les plus proches se rencontrent [MICHALIK et al., 1998]. Le calcul de ce pas en temps est effectué à partir de l'approximation à une dimension de l'équation 1.32, ceci dans le but d'obtenir une fonction de répartition plus facilement inversable par rapport au temps. L'approximation est présentée ci-dessous en équation 1.33 [BERG, 1993].

$$p(x, t) = \frac{1}{\sqrt{4\pi D t}} \cdot e^{-\frac{x^2}{4Dt}} \quad (1.33)$$

avec D le coefficient de diffusion de la molécule considérée dans ce cas précis. La fonction de répartition correspondante est alors :

$$\begin{aligned} p(x \leq r) &= \int_0^r p(x, t) dx \\ &= \left[\operatorname{erf} \left(\frac{x}{2\sqrt{Dt}} \right) \right]_0^r \\ &= \operatorname{erf} \left(\frac{r}{2\sqrt{Dt}} \right) \end{aligned} \quad (1.34)$$

Sachant que :

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-s^2} ds$$

définissons à présent d_0 comme la distance initiale séparant les molécules A et B, R la distance de réaction associée et d_{Amin} et d_{Bmin} les distances minimales qu'elles doivent parcourir pour se rencontrer pendant un temps t_{min} . Ainsi, pour qu'A et B se rencontrent, il faut avoir :

$$d_0 - R = d_{Amin} + d_{Bmin} \quad (1.35)$$

L'objectif est à présent d'introduire la variable temps dans l'équation 1.35 de manière à l'isoler. Dans ce but, on décide de fixer la probabilité qu'une molécule ait parcourue une distance d_{min} a 95% pour pouvoir écrire d'après l'équation 1.34 :

$$p(x \leq d_{min}) = 95$$

et comme :

$$\text{erf}(\sqrt{2}) \approx 0.95$$

On obtient :

$$\begin{aligned} 0.95 = \text{erf}\left(\frac{d_{min}}{2\sqrt{D}t_{min}}\right) &\Leftrightarrow \frac{d_{min}}{2\sqrt{D}t_{min}} = \sqrt{2} \\ &\Leftrightarrow d_{min} = 2\sqrt{2Dt_{min}} \end{aligned} \quad (1.36)$$

En insérant l'équation 1.36 dans l'équation 1.35, on a :

$$\begin{aligned} d_0 - R &= 2\sqrt{2D_A t_{min}} + 2\sqrt{2D_B t_{min}} \\ \Leftrightarrow t_{min} &= \frac{(d_0 - R)^2}{8 \cdot (\sqrt{D_A} + \sqrt{D_B})^2} \end{aligned} \quad (1.37)$$

L'équation 1.37 permet de calculer un pas en temps pour chaque paire de molécules, ce que le code effectue dans un premier temps. Dans un second temps, le code sélectionne le pas en temps le plus court parmi ceux préalablement calculés. Le pas en temps ainsi sélectionné est finalement effectué dans la simulation. La durée d'un pas en temps simulé dépend ainsi de la distribution spatiale des molécules et évolue avec elle. Ainsi, plus la distribution est dispersée, plus les pas en temps calculés sont importants et, à l'inverse, plus la distribution est condensée, plus les pas en temps sont petits. Il reste cependant un élément à calculer pour pouvoir véritablement utiliser l'équation 1.37 : la valeur de R pour chaque couple de molécules considéré. Cette valeur est obtenue à partir d'un résultat de la théorie de Smoluchowski [KARAMITROS et al., 2014] :

$$k = 4\pi \cdot N_a \cdot D \cdot R_r$$

où k est le coefficient de réaction ($[mol]^{-1}[longueur]^3[temps]^{-1}$), N_a le nombre d'Avogadro et R_r le rayon de réaction recherché. Ainsi, sachant que :

$$R = 2 \cdot R_r$$

Il est possible d'écrire pour chaque paire de molécules :

$$R = \frac{k}{2\pi N_a D} \quad (1.38)$$

Les coefficients de réactions disponibles dans Geant4-DNA sont présentés dans le tableau 1.5.

La méthode du "Pont Brownien"

Geant4-DNA propose par défaut une série de restrictions concernant l'amplitude des pas en temps calculés à partir de l'équation 1.37. En effet, tel que nous avons présenté le système jusqu'à maintenant, peu importe qu'il existe une vaste majorité de molécules très espacées dans la simulation car il suffit de deux uniques molécules suffisamment

TABLEAU 1.5 – Les coefficients de réactions pris en compte dans Geant4-DNA [KARAMITROS et al., 2011].

Réaction	Coefficient de réaction ($10^{10} \text{M}^{-1} \text{s}^{-1}$)
$\text{H}^\bullet + \text{e}_{aq}^- + \text{H}_2\text{O} \rightarrow \text{OH}^- + \text{H}_2$	2.65
$\text{H}^\bullet + \text{OH}^\bullet \rightarrow \text{H}_2\text{O}$	1.44
$\text{H}^\bullet + \text{H}^\bullet \rightarrow \text{H}_2$	1.20
$\text{H}_2 + \text{OH}^\bullet \rightarrow \text{H}^\bullet + \text{H}_2\text{O}$	4.17×10^{-3}
$\text{H}_2\text{O}_2 + \text{e}_{aq}^- \rightarrow \text{OH}^- + \text{OH}^\bullet$	1.41
$\text{H}_3\text{O}^+ + \text{e}_{aq}^- \rightarrow \text{H}^\bullet + \text{H}_2\text{O}$	2.11
$\text{H}_3\text{O}^+ \text{OH}^- \rightarrow 2\text{H}_2\text{O}$	14.3
$\text{OH}^\bullet + \text{e}_{aq}^- \rightarrow \text{OH}^\bullet$	2.95
$\text{OH}^\bullet + \text{OH}^\bullet \rightarrow \text{H}_2\text{O}_2$	0.44
$\text{e}_{aq}^- \text{e}_{aq}^- + 2\text{H}_2\text{O} \rightarrow 2\text{OH}^- + \text{H}_2$	0.50

proches pour imposer un très petit pas en temps. Bien que fonctionnelle, cette approche est peu efficace et c'est pourquoi les limites "artificielles" du tableau 1.6 sont mises en place. Ainsi, lorsque le temps de la simulation se situe dans un des intervalles de la première colonne du tableau 1.6, alors les pas en temps ne peuvent être inférieurs à la durée spécifiée dans la seconde colonne du même tableau. Le fait qu'une réaction entre deux molécules très proches puisse être "oubliée" lors de l'application forcée d'un pas en temps minimal est pris en compte au moyen du calcul de probabilité de réaction spécifique. Il s'agit de la technique dite du "Pont Brownien" [PIMBLOTT et LAVERNE, 1992].

TABLEAU 1.6 – La configuration des pas en temps dans Geant4-DNA [KARAMITROS et al., 2011; KREIPL et al., 2008].

Intervalle de temps (s)	Pas en temps minimal (ps)
$< 10^{-11}$	0.1
$10^{-11} - 10^{-10}$	1
$10^{-10} - 10^{-9}$	3
$10^{-9} - 10^{-8}$	10
$> 10^{-8}$	100

Le pont Brownien a pour objectif le calcul de la probabilité d'occurrence d'un événement ayant lieu quelque part sur la trajectoire (Brownienne) d'une molécule. Notons, qu'il s'agit d'un processus stochastique et continu. L'événement d'intérêt est, dans un contexte Geant4-DNA, la réaction entre deux molécules dont la probabilité d'occurrence P_{Br} est donnée par l'équation 1.39. Cette probabilité est toutefois approximative car le problème est, encore une fois, ramené à une dimension pour des raisons pratiques. Le calcul menant à cette expression peut être trouvé dans la littérature [KARAMITROS et al., 2014].

$$P_{Br}(dt) = \exp \frac{-(x_f - R)(x_0 - R)}{D_{A+B} \cdot dt} \quad (1.39)$$

avec D_{A+B} la somme des coefficients de diffusion des deux molécules considérées, R la distance de réaction, x_0 la distance séparant les molécules avant le pas en temps dt et x_f la distance de séparation après ce pas en temps.

1.3.3 Utiliser le module chimie de Geant4-DNA

La création d'une simulation faisant appel au module chimie de Geant4-DNA est très similaire à la création d'une simulation Geant4-DNA classique. Il s'agit de créer une application utilisateur à partir d'un squelette de base que l'utilisateur peut étoffer. Comme le module chimie peut être utilisé au sein d'une application simulant successivement les étapes physique, physico-chimique et chimique ou au sein d'une application uniquement focalisée sur l'étape chimique, il existe deux approches pour construire une application utilisateur :

- Intégrer le module chimie dans une application originalement dédiée à la simulation de l'étape physique. Cela résulte en un programme capable de simuler les trois étapes (physique, physico-chimique et chimique) successivement. Il s'agit de l'approche la plus commune et elle est donc présentée dans les exemples *chem1*, *chem2* et *chem3* fournis avec le code source de Geant4 à partir de la version 10.
- Créer une application uniquement dédiée à la simulation de l'étape chimique. Ce programme est relativement simple car il ne contient que les éléments requis par le module chimie de Geant4-DNA (pas de physique).

Activer le module chimie pour la simulation des étapes physique, physico-chimique et chimique

Si l'utilisateur souhaite activer la chimie pour effectuer une simulation de la radiolyse de l'eau allant de l'irradiation (physique) aux réactions chimiques qui en découlent alors la procédure standard consiste à construire d'abord une application classique dédiée à la simulation de l'étape physique. Cette application est similaire à celle décrite en section 1.2.4. Une fois cette application construite les éléments à modifier pour activer la chimie sont les suivants :

- Une **PhysicsList** adéquate doit être sélectionnée dans le fichier source *PhysicsList.cc*. Cette *PhysicsList* doit contenir non seulement une liste des processus et modèles physiques à utiliser lors de la simulation de l'étape physique mais aussi la liste de ceux nécessaires à la simulation des étapes physico-chimique et chimique. Depuis la version 10.02 de Geant4, une telle *PhysicsList* est fournie directement dans le code source de Geant4 sous le nom de *G4EmDNAChecker*. Dans les versions précédentes, cette *PhysicsList* n'était pas présente et devait être manuellement ajoutée à l'application utilisateur. A cet effet, un modèle était fourni dans chacun des exemples *chem1*, *chem2* et *chem3*.
- **Initialiser le module chimie** dans le *main.cc*.
- Créer une nouvelle classe nommée **StackingAction** (*StackingAction.cc* et *.hh*) et y inclure une méthode *NewStage()* pour démarrer le module chimie une fois l'étape physique terminée.
- Créer une nouvelle classe nommée **TimeStepAction** (*TimeStepAction.cc* et *.hh*) dans laquelle les intervalles de temps sont associés à des pas en temps minimaux. C'est également dans cette classe que se trouvent les méthodes personnalisable par l'utilisateur *UserPreTimeStepAction()* et *UserPostTimeStepAction()* respectivement appelées avant et après chaque pas en temps. La méthode *UserReactionAction()* est appelée après chaque réaction et est également personnalisable. Elle donne accès aux réactifs et produits de chaque réaction et permet à l'utilisateur de les enregistrer dans un fichier durant la simulation.

- Éditer le **ActionInitialization** (*ActionInitialization.cc*) pour y inclure le `StackingAction` et le `TimeStepAction`.

Activer le module chimie pour la simulation de l'étape chimique uniquement

La préparation d'une simulation ne faisant appel qu'au module chimie de Geant4-DNA est relativement aisée. Il s'agit de définir, au minimum, les éléments suivants :

- Un fichier *CMakeLists.txt* pour configurer l'application.
- Un main (*main.cc*) similaire à celui précédemment décrit mais dans lequel, pour démarrer la simulation, on ajoutera :

```
1 G4DNAChemistryManager::Instance() ->Run();
```

- Un `ActionInitialization` (*ActionInitialization.cc* et *.hh*) similaire à celui précédemment décrit mais sans les éléments spécifiques à l'étape physique, c'est à dire le `SteppingAction`, le `TrackingAction`, le `EventAction` et le `RunAction`.
- Une `PhysicsList` (*PhysicsList.cc* et *.hh*) dans laquelle la liste *G4EmDNAChemistry* est sélectionnée.
- Un `DetectorConstruction` où seul le volume racine (`World`) est défini.

Finalement, il est nécessaire d'ajouter les molécules une à une dans la simulation au moyen de la classe *MolecularGun* prévue à cet effet. Cet ajout peut avoir lieu dans le *main.cc* juste avant l'appel de la fonction *Run()*.

1.4 L'ADN et les dommages radio-induits

Une cellule est, en biologie, l'unité fonctionnelle de base du vivant. Elle est constituée d'un noyau situé au sein du cytoplasme, ce dernier étant un milieu occupé par une substance nommée cytosol et différents organites. Chaque organite est un élément spécialisé et interne à la cellule. Il existe deux grands types de cellules : les procaryotes et les eucaryotes. Les procaryotes sont des organismes unicellulaires ne possédant pas de noyau. Les eucaryotes peuvent former des organismes aussi bien uni-cellulaires que multi-cellulaires et sont caractérisées par la présence dans le cytoplasme d'un (ou plusieurs dans certains cas) organite spécifique : le noyau cellulaire. Le noyau est entouré d'une double enveloppe remplie par un liquide visqueux et similaire au cytosol appelé nucleoplasme. En plus du nucleoplasme, le noyau contient divers éléments tels que des protéines, le nucleolus, les corps de Cajal et, entre autres, l'**Acide DésoxyriboNucléique** (ADN). L'ADN contient l'information génétique (ou génome) à l'origine du développement et du fonctionnement des êtres vivants. Les cellules humaines sont en majorité dites "diploïdes", ce qui signifie que leur matériel génétique (ADN) est présent en deux exemplaires. Les cellules n'en possédant qu'un exemplaire sont dites "haploïdes" ce qui est le cas des gamètes par exemple. Le lecteur est invité à se référer aux travaux disponibles dans la littérature pour des informations complémentaires au sujet du cytoplasme [SHEPHERD, 2006], du noyau [HETZER, 2010; HORTON, 2006] et du nucleoplasme [CLEGG, 1984]. La quantité d'ADN présente dans une cellule humaine diploïde est équivalente à un filament d'ADN d'environ deux mètres de longueur. La façon dont cette quantité d'ADN est répartie dans le noyau cellulaire est encore aujourd'hui un sujet de recherche à part entière. En effet, la structure de l'ADN est à la fois multi-échelle et dynamique [LUGER et al., 2012] ce qui rend son étude particulièrement complexe tant expérimentalement que théoriquement.

[HARP et al., 2000] et illustrée en figure 1.14. Les nucléosomes ainsi formés sont reliés entre eux par une prolongation de la double hélice : le linker. Ainsi, la succession de plusieurs nucléosomes donne lieu à la création d'une chaîne ADN (double hélice) ininterrompue appelée chromatine. Les nucléosomes de cette chaîne sont spatialement distribués de manière à former une fibre avec environ 200 paires de nucléotides par nucléosome. Les caractéristiques de cette fibre semblent dépendre de nombreux facteurs biologiques [BRANCO et POMBO, 2007; HANSEN, 2012; LUGER et al., 2012] et ne font pas l'objet d'un consensus ferme. D'autant que les données et analyses effectuées *in vitro* ne sont pas forcément applicables à la distribution de la chromatine *in vivo* [HANSEN, 2012; TREMETHICK, 2007]. Toutefois, il existe deux modèles particulièrement mis en avant pour décrire l'organisation des nucléosomes à l'intérieur d'une fibre de chromatine : le modèle "solénoïde" et le modèle "zig-zag" qui sont illustrés en figure 1.15. Le diamètre de la fibre de chromatine est également souvent présenté comme valant 30 nm.

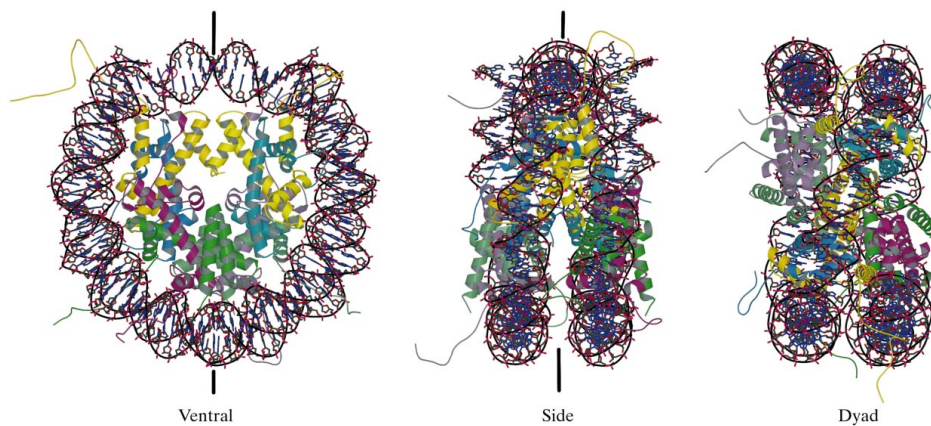


FIGURE 1.14 – Vues d'un nucléosome avec la double hélice de l'ADN enroulée autour du complexe de protéines histones [HARP et al., 2000].

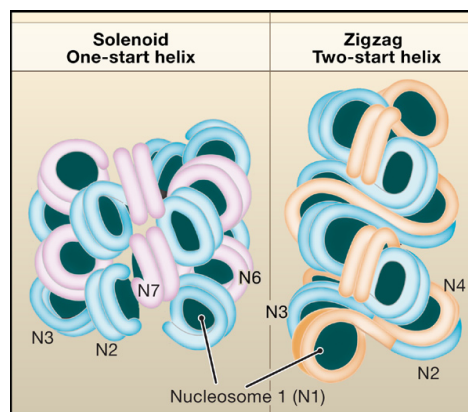


FIGURE 1.15 – Organisation des nucléosomes au sein d'une fibre de chromatine selon les modèles "solénoïde" et "zigzag" [TREMETHICK, 2007].

1.4.2 Distribution spatiale de la chromatine dans le noyau cellulaire

A l'intérieur d'un noyau de cellule humaine diploïde, il existe 46 chromosomes (23 paires) constitués chacun d'une fibre de chromatine. La répartition des fibres de chromatines au sein du noyau est liée au cycle cellulaire représenté schématiquement en figure

1.16. En effet, lors de la division cellulaire (mitose) l'ADN est compacté, dupliqué et transmis aux cellules filles [RAYNAUD et al., 2014] :

- État quiescent (G0) : la cellule n'est pas en train de se diviser et est dans un état de "repos" du point de vue de la mitose. Précisons que ce "repos" est relatif car la cellule fonctionne tout de même et l'ADN est donc en permanence lu et transcrit.
- Interphase (G1, S, G2) : la cellule se prépare en augmentant sa taille (G1), puis l'ADN est dupliqué (S) et la cellule continue de grossir.
- Prophase (début de la mitose) : la chromatine se condense pour former des chromosomes bien distincts. En raison de la duplication du matériel génétique ayant eu lieu dans l'interphase, la condensation des chromosomes fait apparaître deux brins (chromatides) pour chaque chromosome. Les brins sont reliés deux à deux donnant ainsi aux chromosomes une forme de croix.
- Métaphase : les chromosomes condensés se rassemblent à l'équateur de la cellule et s'alignent selon leur centromère.
- Anaphase : les chromatides se séparent et se dirigent vers les pôles opposés de la cellule.
- Télaphase (fin de la mitose) : les chromatides se décondensent et les cellules filles se séparent progressivement.

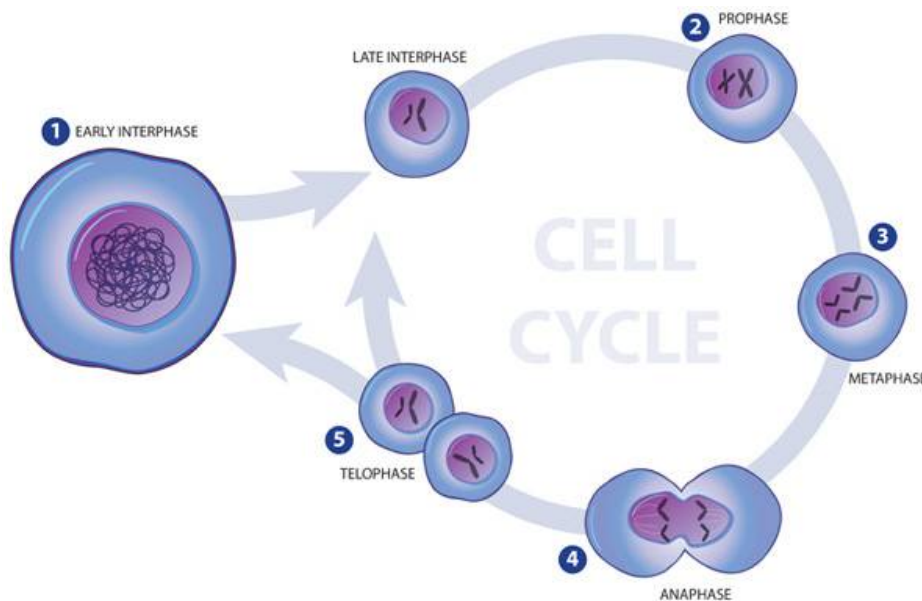


FIGURE 1.16 – Les différentes structures de la chromatine lors du cycle cellulaire. "Early interphase" correspond à la phase G0. (source : www.nature.com/scitable/content).

Finalement le degré de compaction de la chromatine varie significativement lors du cycle cellulaire et cela implique des changements au niveau de l'organisation spatiale de la chromatine. Lorsque la chromatine est dans un état condensé durant la fin de la prophase et la métaphase alors sa structure n'est pas considérée comme connue bien que différents modèles aient été proposés [COOK, 1995; HANSEN, 2012; TAKAHASHI, 1989]. Une organisation simplifiée est proposée en figure 1.17 (cf. échelle 700 nm) pour résumer l'ensemble des niveaux structurels préalablement présentés. Dans le cas de la phase G0 du cycle cellulaire, la chromatine est dans un état décondensé et sa structure est plus accessible expérimentalement [COSTANTINI et al., 2006; CREMER et CREMER, 2001; RAO et al., 2014]. Cependant, les résultats des expériences sont souvent très spécifiques et avec

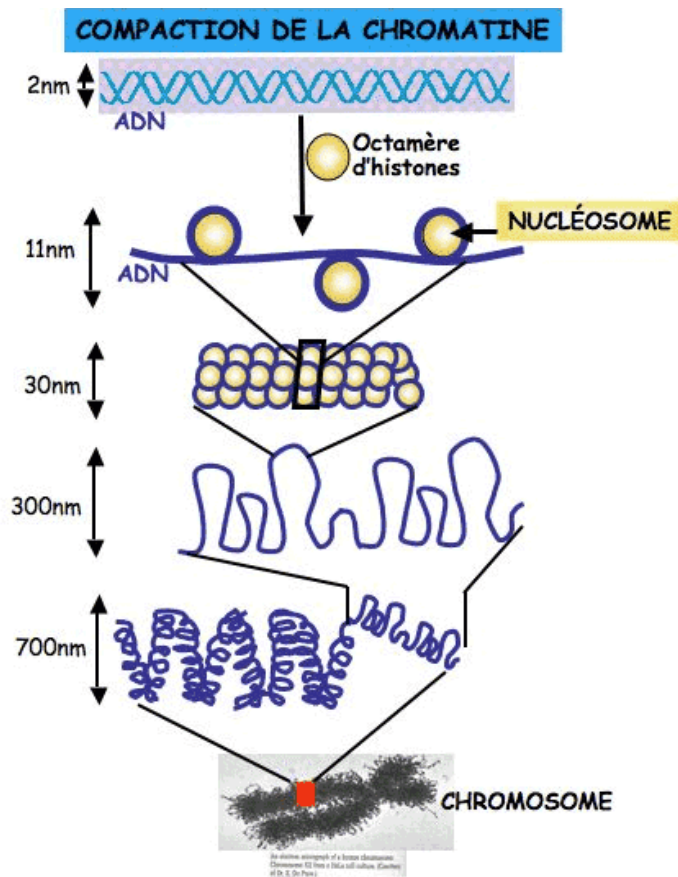


FIGURE 1.17 – Les différents niveaux de compaction de la chromatine (source : <http://www.facbio.com>).

une résolution limitée expliquant ainsi l'existence, dans ce cas également, de nombreux modèles proposant des organisations spatiales différentes de la chromatine [BANCAUD et al., 2012; BRANCO et POMBO, 2007]. Une organisation fractale de la chromatine a été récemment mise en avant suite à la publication d'une nouvelle carte du génome humain [LIEBERMAN-AIDEN et al., 2009; RAO et al., 2014; SCHRAM et al., 2013].

Les différents modèles d'organisation de la chromatine dans le noyau partagent toutefois certains éléments communément acceptés. Ainsi en G₀, la fibre de chromatine effectue une série de boucles présentes par paquets dans des zones sphériques, les "domaines". Ces domaines sont eux mêmes répartis au sein des entités de plus haut niveau que sont les territoires chromosomiques. Il existe un territoire chromosomique par chromosome, soit un total de 46 territoires pour une cellule humaine diploïde. La fibre de chromatine n'est pas toujours identique au sein de ces territoires et une distinction est effectuée entre les zones où la fibre est dense (nucléosomes rapprochés) formant l'hétérochromatine et les zones où la fibre est plus espacée (nucléosomes éloignés) qui constituent l'euchromatine [DILLON, 2004]. Quantitativement, un domaine peut contenir un ou plusieurs millions de paires de nucléotides tandis qu'un territoire chromosomique peut rassembler plusieurs centaines de domaines. Finalement, une cellule diploïde humaine contient environ 6,4 milliards de paires de nucléotides dans son noyau.

1.4.3 Les dommages à l'ADN

Les dommages radio-induits sur l'ADN peuvent être mineurs ou majeurs. Les dommages majeurs induisent une mort cellulaire immédiate (nécrose) ou différée (apoptose). Les dommages mineurs sont réparables mais cette réparation n'est pas forcément exacte. Lorsque la réparation est inexacte alors la séquence de gènes est altérée et le patrimoine génétique de la cellule est modifié : c'est une mutation génique. Les dommages radio-induits à l'ADN sont générés lors des étapes physique et chimique et sont généralement catégorisés en cassures simples ou doubles brin (SSB et DSB), dommages de base, pontages et lésions multiples localisées. Les cassures double brin sont connues comme étant particulièrement critiques [GOODHEAD, 1994; OLIVE, 1998] et leur modélisation fait l'objet de ce travail de thèse.

Cassures de brin

Les cassures de brins sont généralement localisées au niveau du squelette ("backbone") de l'ADN (phosphate et 2-désoxyribose) et elles sont causées par une rupture de la liaison phosphate-désoxyribose ou, plus fréquemment, de la liaison base-désoxyribose. On distingue deux types de cassures de brin : les cassures simple brin et les cassures double brin. Les cassures simple brin sont les plus courantes et les plus facilement réparables par la cellule. Leur influence sur la survie cellulaire est donc faible. Parmi les cassures simple brin, les trois catégories de la figure 1.18 sont communément définies [NIKJOO et al., 1997]. Les cassures double brin sont moins courantes et définies à partir d'au moins deux ruptures de l'ADN. Ces ruptures doivent être situées sur deux brins opposés et distantes de moins de 10 paires de nucléotides. La facilité avec laquelle la cellule les répare diminue avec leur complexité, c'est à dire avec le nombre de ruptures contenues dans une cassure double brin. Une rupture étant causée par une cassure simple brin. Les différents types de cassures double brin sont présentés en figure 1.19 [NIKJOO et al., 1997].




Caractéristique	Représentation	Nom
Dépôt d'énergie localisé à une distance supérieure à 10 bp de tout autre point		SSB
Deux dépôts d'énergie ou plus localisés sur le même brin à une distance inférieure à 10 bp		SSB+
Deux dépôts d'énergie ou plus localisés sur les deux brins à une distance supérieure à 10 bp		SSBs

FIGURE 1.18 – Cassures simple brin de l'ADN [DOS SANTOS, 2013].




Caractéristique	Représentation	Nom
Deux dépôts d'énergie localisés sur les deux brins à une distance inférieure à 10 bp		DSB
Plus de deux dépôts d'énergie localisés sur les deux brins à une distance inférieure à 10 bp		DSB+
Plusieurs DSB séparées par une distance inférieure à 10 bp		DSB++

FIGURE 1.19 – Cassures double brin de l'ADN [DOS SANTOS, 2013].

Autres types de dommages

Les cassures de brin ne sont pas les seuls types de dommages radio-induits que peut subir l'ADN. Il existe également les dommages de bases, les pontages et les lésions multiples localisées.

- Les dommages de bases correspondent à l'endommagement de bases lors de l'étape physique ou chimique. Un tel dommage peut avoir lieu suite à une ionisation dans une base ou, plus couramment, suite à une réaction avec le radical OH^\bullet . Les différentes bases réagissent plus ou moins facilement avec ce radical et peuvent donc être classées par ordre de radiosensibilité : $\text{T} > \text{C} \gg \text{A} > \text{G}$ [AVERBECK, 2008]. Dans l'ensemble, deux à trois dommages de bases sont identifiés pour dix cassures simple brin [AVERBECK, 2008].
- Les radicaux OH^\bullet peuvent altérer la structure de l'ADN et provoquer l'apparition d'un site abasique. Il s'agit d'une paire de nucléotide dont une des deux bases est manquante malgré le fait que le backbone soit ininterrompu. Un tel dommage est relativement rare mais peut se transformer en cassure simple brin en l'absence de réparation.
- Les pontages sont des liaisons ("ponts") qui peuvent apparaître entre deux chaînes ADN distinctes (inter-chaîne) ou au sein d'une même chaîne ADN (intra-chaîne). Notons que la présence de pontages ADN-protéine est également possible. Les pontages perturbent la séquence de gènes ainsi que les différents mécanismes de réparation et lecture de l'ADN. Un pontage suit généralement une réparation ratée de la chaîne ADN et cette réparation peut succéder à un dommage radio-induit.
- Les lésions multiples localisées sont des cassures simple ou double brin associées à des dommages de bases distants de moins de 20 paires de nucléotides.

Détection expérimentale des cassures double brins

Les cassures double brin de l'ADN que ce travail cherche à calculer au moyen d'une simulation mécanistique sont expérimentalement détectables par différentes techniques dont l'immunofluorescence γH2AX fait partie. Il s'agit d'une technique basée sur le fait que le complexe de protéines histones mentionné en section 1.4.1 est constitué de 4 protéines centrales ("core") et d'une protéine ("link") pour stabiliser et lier les nucléosomes entre eux. Les 4 protéines centrales sont H2A, H2B, H3 et H4. La protéine H2A a ceci de particulier qu'il en existe une variante H2AX que l'on retrouve dans 2 à 25% des histones du noyau cellulaire. Cette protéine a également pour caractéristique d'être uniformément répartie dans le noyau cellulaire permettant ainsi son utilisation pour la détection des cassures double brin. En effet, une cassure double brin ouvre un accès à cette protéine au sein du nucléosome et un anticorps spécifique peut s'y glisser puis se fixer à la protéine H2AX préalablement phosphorylée. Ces anticorps contiennent une substance fluorochrome qui les rend visible au microscope permettant ainsi de localiser une partie des cassures double brin. Les tâches lumineuses observées au microscope et provenant de la fixation de ces anticorps à la protéine H2AX phosphorylées sont appelés des foci. La figure 1.20 présente deux images contenant des foci. La première image est issue d'une irradiation effectuée avec des neutrons qui mettent en mouvement des protons de recul tandis que la seconde provient d'une irradiation avec des photons. L'organisation spatiale des foci est différente d'une image à l'autre illustrant ainsi les impacts divers que peut avoir l'irradiation d'une cellule avec différentes particules.

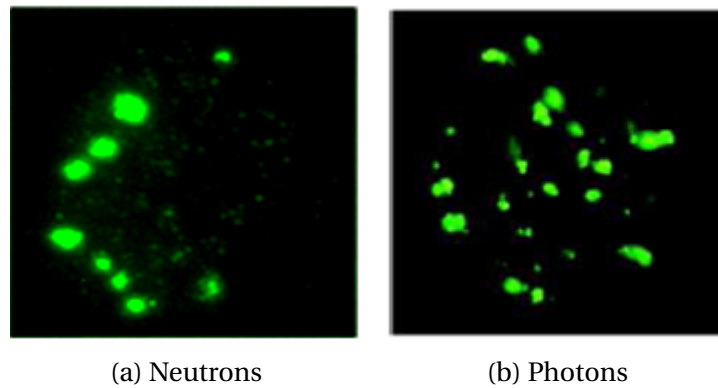


FIGURE 1.20 – Les cassures double brin de l'ADN détectées avec les foci formés suite à l'irradiation d'une cellule de type fibroblastique [DOS SANTOS, 2013].

1.4.4 Modélisations de l'ADN pour le calcul des dommages radio-induits

Le calcul des dommages radio-induits à l'ADN nécessite d'introduire une représentation de l'ADN dans le calcul. Cette représentation peut être de différentes natures : basée sur des concentrations où, au contraire, très détaillée géométriquement. Dans le cadre de la simulation des dommages radio-induits avec une approche mécanistique, il est nécessaire de localiser les dommages à l'ADN à l'échelle nanométrique et, donc, les représentations géométriques utilisées doivent avoir une résolution similaire. Cependant, même en fixant la résolution à quelques nanomètres, il existe de multiples choix de représentation possibles qui ont engendré de nombreuses modélisations de l'ADN. Ainsi, en fonction des besoins, l'ADN peut être représenté par un simple cylindre [LAZARAKIS et al., 2012] ou, dans d'autres situations, par une géométrie définie atome par atome. La figure 1.21 illustre ces deux représentations. Les représentations les plus simples ayant pour avantage d'être légères, facilement manipulables et peu coûteuses en termes de temps de calcul et taille de fichier. Elles correspondent également souvent aux représentations utilisées au sein de dispositifs expérimentaux [GARTY et al., 2010; PSZONA et al., 2000]. En contrepartie, ces représentations limitent la capacité de localisation des dommages à l'ADN dans une simulation. En effet, un calcul effectué avec une représentation cylindrique ne permettra pas d'identifier quelles molécules de l'ADN ont été ionisées mais plutôt quel a été le nombre moyen d'ionisations dans un volume donné. De plus, la différence occasionnée par l'utilisation de différents modèles géométriques sur la complexité des dommages à l'ADN a été étudiée par [BUENO et al., 2015]. Ce volume pouvant être équivalent à celui occupé par une dizaine de paires de nucléotides par exemple. Les représentations très détaillées [BERNAL et al., 2013; FRIEDLAND et al., 2011; POMPLUN et TERRISSOL, 1994] permettent de localiser très précisément les différents dommages à l'ADN durant une simulation mais au prix d'un temps de calcul plus élevé et d'une flexibilité moindre. En effet, les modèles géométriques ainsi construits sont complexes et de taille conséquente ce qui rend leur modification et adaptation à des situations spécifiques délicates. A fortiori lorsque ces modèles détaillent l'ensemble du génome contenu dans un noyau cellulaire tel qu'illustré en figure 1.22 [FRIEDLAND et KUNDRÁT, 2013; NIKJOO et GIRARD, 2012].

Dans l'ensemble, le choix d'un modèle d'ADN dépend du résultat attendu et du type de simulation menée. Si l'utilisation des modèles les plus détaillés est souvent tentante, il est utile de garder à l'esprit leurs inconvénients en terme de performance. Leur utilisation soulève également certaines questions fondamentales liées aux limites des simulations de Monte Carlo, notamment physique. En effet, la localisation précise d'une interaction

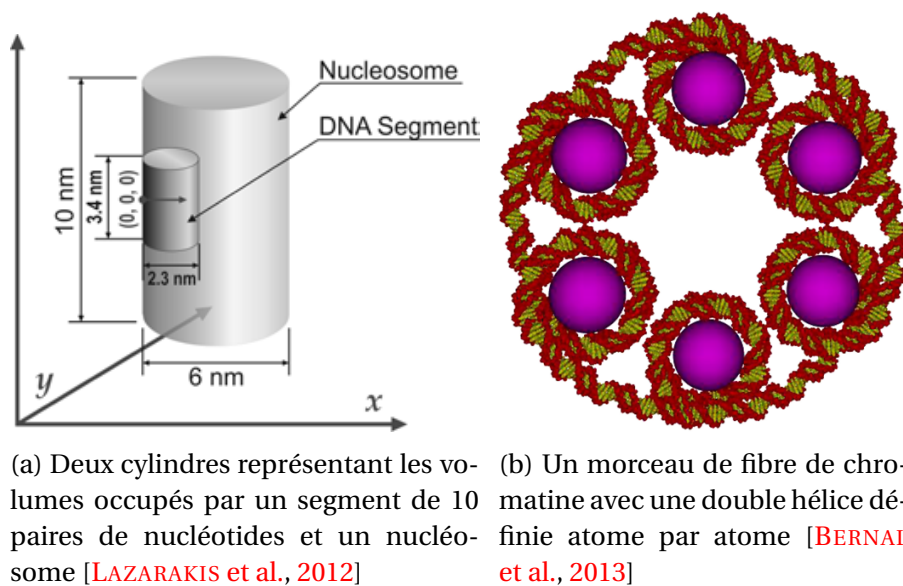


FIGURE 1.21 – Deux représentations de l'ADN avec des niveaux de détails différents.

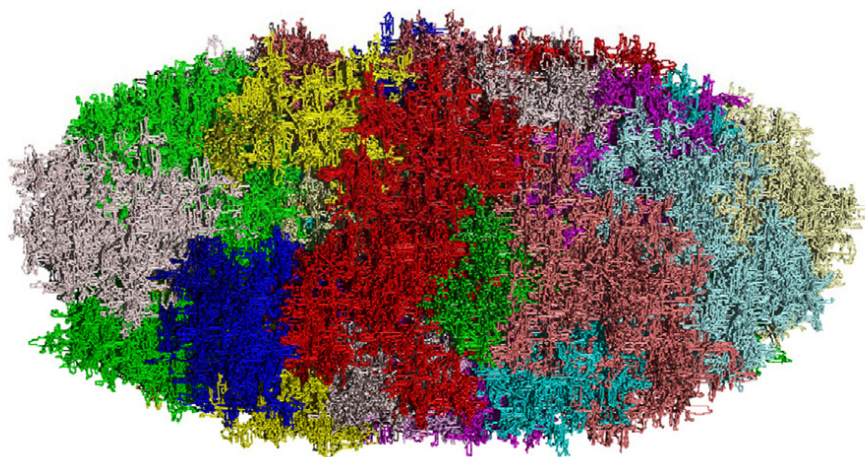


FIGURE 1.22 – Modèle géométrique de l'ADN contenu dans un noyau de cellule fibroblaste (visualisation : raytracing) [FRIEDLAND et KUNDRÁT, 2013].

physique à l'échelle du nanomètre au sein d'une simulation de Monte Carlo classique ne prend pas en compte certains aspects quantiques, tel que le principe d'incertitude d'Heisenberg, dont l'influence peut devenir significative lorsque des particules de très basses énergies (< 1keV) sont transportées [THOMSON et KAWRAKOW, 2011]. Le mode de fonctionnement classique des codes de Monte Carlo équivaut dans ce cas à une approximation qu'il convient de considérer. Une autre caractéristique de ces modèles ADN détaillés est leur aspect statique. En effet, l'ADN y est toujours immobile et le cycle cellulaire figé ce qui n'est évidemment pas le cas des conditions *in vivo* réelles.

Références

- AGOSTINELLI, S., J. ALLISON, K. AMAKO, J. APOSTOLAKIS, H. ARAUJO, P. ARCE, M. ASAI, D. AXEN, S. BANERJEE, G. BARRAND et E. AL. 2003, «Geant4 - A simulation toolkit», *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 506, n° 3, p. 250–303. [15](#), [17](#)
- ALLEN, A. O. 1961, *The radiation chemistry of water and aqueous solutions*, van Nostrand New York. [29](#)
- ALLISON, J., K. AMAKO, J. APOSTOLAKIS, H. ARAUJO, P. A. DUBOIS, M. ASAI, G. BARRAND, R. CAPRA, S. CHAUVIE, R. CHYTRACEK et al.. 2006, «Geant4 developments and applications», *Nuclear Science, IEEE Transactions on*, vol. 53, n° 1, p. 270–278. [15](#), [17](#)
- ARNOTT, S. 2006, «Historical article : DNA polymorphism and the early history of the double helix», *Trends in Biochemical Sciences*, vol. 31, n° 6, p. 349–354. [39](#)
- AVERBECK, D. 2008, *Radiobiologie : radiothérapie et radioprotection, bases fondamentales.*, hermann éd.. [45](#)
- BALDACCHINO, G. et B. HICKEL. 2008, «Water radiolysis under extreme conditions. Application to the nuclear industry», *Radiation chemistry. EDP Sciences, France*, p. 53–54. [29](#), [30](#)
- BALLINGER, C. T., D. E. CULLEN, S. T. PERKINS, J. A. RATHKOPF, W. R. MARTIN et S. J. WILDERMAN. 1992, «Single-scatter Monte Carlo compared to condensed history results for low energy electrons», *Nuclear Instruments and Methods in Physics Research Section B : Beam Interactions with Materials and Atoms*, vol. 72, n° 1, p. 19–27. [17](#)
- BANCAUD, A., C. LAVELLE, S. HUET et J. ELLENBERG. 2012, «A fractal model for nuclear organization: current evidence and biological implications», *Nucleic Acids Research*, vol. 40, n° 18, p. 8783–8792. [43](#)
- BERG, H. C. 1993, *Random walks in biology*, Princeton University Press. [34](#)
- BERNAL, M., M. BORDAGE, J. BROWN, M. DAVIDKOVÁ, E. DELAGE, Z. EL BITAR, S. ENGER, Z. FRANCIS, S. GUATELLI, V. IVANCHENKO, M. KARAMITROS, I. KYRIAKOU, L. MAIGNE, S. MEYLAN, K. MURAKAMI, S. OKADA, H. PAYNO, Y. PERROT, I. PETROVIC, Q. PHAM, A. RISTIC-FIRA, T. SASAKI, V. ŠTĚPÁN, H. TRAN, C. VILLAGRASA et S. INCERTI. 2015, «Track structure modeling in liquid water: A review of the Geant4-DNA very low energy extension of the Geant4 Monte Carlo simulation toolkit», *Physica Medica*, vol. 31, n° 8, p. 861–874. [23](#), [24](#)

- BERNAL, M., D. SIKANSI, F. CAVALCANTE, S. INCERTI, C. CHAMPION, V. IVANCHENKO et Z. FRANCIS. 2013, «An atomistic geometrical model of the B-DNA configuration for DNA–radiation interaction simulations», *Computer Physics Communications*, vol. 184, n° 12, p. 2840–2847. [46](#), [47](#)
- BOICE JR, J. D. 2012, «Radiation epidemiology : a perspective on Fukushima», *Journal of Radiological Protection*, vol. 32, n° 1, p. N33–N40. [12](#)
- BOLUS, N. E. 2001, «Basic review of radiation biology and terminology», *Journal of nuclear medicine technology*, vol. 29, n° 2, p. 67–73. [9](#), [11](#)
- BRANCO, M. R. et A. POMBO. 2007, «Chromosome organization: new facts, new models», *Trends in Cell Biology*, vol. 17, n° 3, p. 127–134. [40](#), [43](#)
- BRUN, R., F. DAMIAN, C. DIOP, E. DUMONTEIL, F. HUGOT, C. JOUANNE, Y. LEE, F. MALVAGI, A. MAZZOLO, O. PETIT, J. TRAMA, T. VISONNEAU et A. ZOIA. 2015, «TRIPOLI4 CEA EDF and AREVA reference Monte Carlo code», *Annals of Nuclear Energy*, vol. 82, p. 151–160. [15](#)
- BUENO, M., R. SCHULTE, S. MEYLAN et C. VILLAGRASA. 2015, «Influence of the geometrical detail in the description of DNA and the scoring method of ionization clustering on nanodosimetric parameters of track structure : a Monte Carlo study using Geant4-DNA», *Physics in Medicine and Biology*, vol. 60, n° 21, p. 8583–8599. [46](#)
- CLEGG, J. S. 1984, «Properties and metabolism of the aqueous cytoplasm and its boundaries», *AM. J. Physiol.*, vol. 246. [38](#)
- COOK, P. R. 1995, «A chromomeric model for nuclear and chromosome structure», *Journal of Cell Science*, vol. 108, n° 9, p. 2927–2935. [41](#)
- COSTANTINI, M., O. CLAY, F. AULETTA et G. BERNARDI. 2006, «An isochore map of human chromosomes», *Genome research*, vol. 16, n° 4, p. 536–541. [41](#)
- CREMER, T. et C. CREMER. 2001, «Chromosome territories, nuclear architecture and gene regulation in mammalian cells», *Nature reviews genetics*, vol. 2, n° 4, p. 292–301. [41](#)
- CUCINOTTA, F. A., F. K. MANUEL, J. JONES, G. ISZARD, J. MURREY, B. DJOJONEGRO et M. WEAR. 2001, «Space radiation and cataracts in astronauts», *Radiation research*, vol. 156, n° 5, p. 460–466. [12](#)
- DIEUDONNÉ, C. 2013, *Accélération de la simulation Monte Carlo du transport des neutrons dans un milieu évoluant par la méthode des échantillons corrélés.*, thèse de doctorat, Université Paris Sud-Paris XI. [15](#)
- DILLON, N. 2004, «Heterochromatin structure and function», *Biology of the Cell*, vol. 96, n° 8, p. 631–637. [43](#)
- DOS SANTOS, M. 2013, *Modélisation de la topologie des dépôts d'énergie créés par un rayonnement ionisant à l'échelle nanométrique dans les noyaux cellulaires et relation avec les événements précoces radio-induits*, thèse de doctorat, Bordeaux 1, Ecole doctorale des sciences physiques et de l'ingénieur. [10](#), [24](#), [44](#), [46](#)
- DOWD, S. et E. TILSON. 1999, *Practical Radiation Protection and Applied Radiobiology*, 2, Saunders, Philadelphia. [11](#), [12](#)

- ERMAK, D. L. et J. MCCAMMON. 1978, «Brownian dynamics with hydrodynamic interactions», *The Journal of Chemical Physics*, vol. 69, p. 1352. [33](#)
- FRANCIS, Z. 2007, *Simulations Monte Carlo et étude microdosimétrique pour des irradiations cellulaires à faibles doses en neutrons de 14 MeV*, thèse de doctorat, Université Blaise Pascal-Clermont-Ferrand II. [24](#)
- FRIEDLAND, W., M. DINGFELDER, P. KUNDRÁT et P. JACOB. 2011, «Track structures, DNA targets and radiation effects in the biophysical Monte Carlo simulation code PARTRAC», *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*, vol. 711, n° 1-2, p. 28–40. [23](#), [46](#)
- FRIEDLAND, W. et P. KUNDRÁT. 2013, «Track structure based modelling of chromosome aberrations after photon and alpha-particle irradiation», *Mutation Research/Genetic Toxicology and Environmental Mutagenesis*, vol. 756, n° 1-2, p. 213–223. [46](#), [47](#)
- GARTY, G., R. SCHULTE, S. SHCHEMELININ, C. LELOUP, G. ASSAF, A. BRESKIN, R. CHECHIK, V. BASHKIROV, J. MILLIGAN et B. GROSSWENDT. 2010, «A nanodosimetric model of radiation-induced clustered DNA damage yields», *Physics in medicine and biology*, vol. 55, n° 3, p. 761. [46](#)
- GOODHEAD, D. 1994, «Initial events in the cellular effects of ionizing radiations : clustered damage in DNA», *Int. J. Radiat. Biol.*, vol. 65, n° 1, p. 7–17. [43](#)
- GOORLEY, T., M. JAMES, T. BOOTH, F. BROWN, J. BULL, L. J. COX, J. DURKEE, J. ELSON, M. FENSIN, R. A. FORSTER, J. HENDRICKS, H. G. HUGHES, R. JOHNS, B. KIEDROWSKI, R. MARTZ, S. MASHNIK, G. MCKINNEY, D. PELOWITZ, R. PRAEL, J. SWEEZY, L. WATERS, T. WILCOX et T. ZUKAITIS. 2012, «Initial MCNP6 Release Overview», *Radiation Transport and Protection*. [15](#)
- HANSEN, J. C. 2012, «Human mitotic chromosome structure : what happened to the 30-nm fibre?», *The EMBO journal*, vol. 31, n° 7, p. 1621–1623. [40](#), [41](#)
- HARP, J. M., B. L. HANSON, D. E. TIMM et G. J. BUNICK. 2000, «Asymmetries in the nucleosome core particle at 2.5 Å resolution», *Acta Crystallographica Section D : Biological Crystallography*, vol. 56, n° 12, p. 1513–1534. [40](#)
- HART, E. J. et J. BOAG. 1962, «Absorption Spectrum of the Hydrated Electron in Water and in Aqueous Solutions», *Journal of the American Chemical Society*, vol. 84, n° 21, p. 4090–4095. [31](#)
- HETZER, M. W. 2010, «The Nuclear Envelope», *Cold Spring Harbor Perspectives in Biology*, vol. 2, n° 3, p. a000 539–a000 539. [38](#)
- HORTON, T. R. 2006, «The number of nuclei in basidiospores of 63 species of ectomycorrhizal Homobasidiomycetes», *Mycologia*, vol. 98, n° 2, p. 233–238. [38](#)
- INCERTI, S., G. BALDACCHINO, M. BERNAL, R. CAPRA, C. CHAMPION, Z. FRANCIS, P. GUÈYE, A. MANTERO, B. MASCIALINO, P. MORETTO et OTHERS. 2010, «The Geant4-DNA project», *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 1, n° 02, p. 157–178. [23](#), [24](#)

- IVANCHENKO, V. N., J. APOSTOLAKIS, A. SCHAELOCKE, L. URBAN, T. TOSHITO, S. ELLES, J. JACQUEMIER, M. MAIRE, A. BAGULYA, V. GRICHINE et OTHERS. 2014, *Geant4 Physics reference Manual*, vol. 108. [20](#)
- KARAMITROS, M. 2012, *Extension de l'outil Monte Carlo généraliste Geant4 pour la simulation de la radiolyse de l'eau dans le cadre du projet Geant4-DNA*, thèse de doctorat, Bordeaux 1, Bordeaux. [24](#), [30](#), [31](#)
- KARAMITROS, M., S. LUAN, M. BERNAL, J. ALLISON, G. BALDACCHINO, M. DAVIDKOVA, Z. FRANCIS, W. FRIEDLAND, V. IVANTCHENKO, A. IVANTCHENKO, A. MANTERO, P. NIEMINEM, G. SANTIN, H. TRAN, V. STEPAN et S. INCERTI. 2014, «Diffusion-controlled reactions modeling in Geant4-DNA», *Journal of Computational Physics*, vol. 274, p. 841–882. [30](#), [34](#), [35](#), [36](#)
- KARAMITROS, M., A. MANTERO, S. INCERTI, W. FRIEDLAND, G. BALDACCHINO, P. BARBERET, M. BERNAL, R. CAPRA, C. CHAMPION, Z. BITAR et OTHERS. 2011, «Modeling radiation chemistry in the Geant4 toolkit», *Progress in nuclear science and technology*, vol. 2, p. 503–508. [30](#), [33](#), [36](#)
- KERNBAUM, M. 1910, «Décomposition de l'eau par l'aigrette», *Le Radium*, vol. 7, n° 9, p. 275–278. [29](#)
- KREIPL, M. S., W. FRIEDLAND et H. G. PARETZKE. 2008, «Time- and space-resolved Monte Carlo study of water radiolysis for photon, electron and ion irradiation», *Radiation and Environmental Biophysics*, vol. 48, n° 1, p. 11–20. [36](#)
- KROESE, D. P., T. BRERETON, T. TAIMRE et Z. I. BOTEV. 2014, «Why the Monte Carlo method is so important today: Why the MCM is so important today», *Wiley Interdisciplinary Reviews : Computational Statistics*, vol. 6, n° 6, p. 386–392. [14](#)
- KRÄMER, M. 2009, «Swift ions in radiotherapy–Treatment planning with TRiP98», *Nuclear Instruments and Methods in Physics Research Section B : Beam Interactions with Materials and Atoms*, vol. 267, n° 6, p. 989–992. [11](#)
- LADOU, J. 1990, *Occupational Medicine*, Appleton & Lange, Norwalk. [12](#)
- LARSEN, E. W. 1992, «A theoretical derivation of the condensed history algorithm», *Annals of Nuclear Energy*, vol. 19, n° 10-12, p. 701–714. [17](#)
- LAZARAKIS, P., M. U. BUG, E. GARGIONI, S. GUATELLI, H. RABUS et A. B. ROSENFELD. 2012, «Comparison of nanodosimetric parameters of track structure calculated by the Monte Carlo codes Geant4-DNA and PTr», *Physics in Medicine and Biology*, vol. 57, n° 5, p. 1231. [46](#), [47](#)
- LESLIE, A., S. ARNOTT, R. CHANDRASEKARAN et R. RATLIFF. 1980, «Polymorphisme of DNA double helices», *J. Mol. Biol.*, vol. 143, p. 49–72. [39](#)
- LIEBERMAN-AIDEN, E., N. L. VAN BERKUM, L. WILLIAMS, M. IMAKAEV, T. RAGOCZY, A. TELLING, I. AMIT, B. R. LAJOIE, P. J. SABO, M. O. DORSCHNER, R. SANDSTROM, B. BERNSTEIN, M. A. BENDER, M. GROUDINE, A. GNIRKE, J. STAMATOYANNOPOULOS, L. A. MIRNY, E. S. LANDER et J. DEKKER. 2009, «Comprehensive Mapping of Long-Range Interactions Reveals Folding Principles of the Human Genome», *Science*, vol. 326, n° 5950, p. 289–293. [43](#)

- LUGER, K., M. L. DECHASSA et D. J. TREMETHICK. 2012, «New insights into nucleosome and chromatin structure : an ordered state or a disordered affair?», *Nature Reviews Molecular Cell Biology*, vol. 13, n° 7, p. 436–447. [38](#), [40](#)
- MELTON, C. E. 1972, «Cross Sections and Interpretation of Dissociative Attachment Reactions Producing OH-, O-, and H- in H₂O», *The Journal of Chemical Physics*, vol. 57, n° 10, p. 4218. [31](#)
- METROPOLIS, N. 1987, «The beginning of the Monte Carlo method», *Los Alamos Science*, vol. 15, n° 584, p. 125–130. [14](#)
- METROPOLIS, N. et S. ULAM. 1949, «The Monte Carlo Method», *Journal of tyhe american statistical association*, vol. 44, n° 247, p. 335–341. [14](#)
- MICHALIK, V., M. BEGUSOVA et E. A. BIGILDEEV. 1998, «Computer-Aided Stochastic Modeling of the Radiolysis of Liquid Water», *Radiation Research*, vol. 149, p. 224–236. [34](#)
- MORRIS, K. V. et J. S. MATTICK. 2014, «The rise of regulatory RNA», *Nature Reviews Genetics*, vol. 15, n° 6, p. 423–437. [39](#)
- MOYSICH, K. B., R. J. MENEZES et A. M. MICHALEK. 2002, «Chernobyl-related ionising radiation exposure and cancer risk : an epidemiological review», *The Lancet Oncology*, vol. 3, n° 5, p. 269–279. [12](#)
- NIKJOO, H. et P. GIRARD. 2012, «A model of the cell nucleus for DNA damage calculations», *International Journal of Radiation Biology*, vol. 88, n° 1-2, p. 87–97. [23](#), [46](#)
- NIKJOO, H., P. O'NEILL, D. GOODHEAD et M. TERRISOL. 1997, «Computational modelling of low-energy electron-induced DNA damage by early physical and chemical events», *International Journal of Radiation Biology*, vol. 71, n° 5, p. 467–483. [43](#)
- NIKJOO, H., S. UEHARA, D. EMFIETZOGLOU et F. CUCINOTTA. 2006, «Track-structure codes in radiation research», *Radiation Measurements*, vol. 41, n° 9-10, p. 1052–1074. [14](#), [18](#)
- OLIVE, P. L. 1998, «The Role of DNA Single- and Double-Strand Breaks in Cell Killing by Ionizing Radiation», *Radiation Research*, vol. 150, p. 42–51. [14](#), [43](#)
- PEDERSEN, J. S., G. BEJERANO, A. SIEPEL, K. ROSENBLOOM, K. LINDBLAD-TOH, E. S. LANDER, J. KENT, W. MILLER et D. HAUSSLER. 2006, «Identification and Classification of Conserved RNA Secondary Structures in the Human Genome», *PLoS Computational Biology*, vol. 2, n° 4, p. e33. [39](#)
- PIERCE, D. A. et D. L. PRESTON. 2000, «Radiation-related cancer risks at low doses among atomic bomb survivors», *Radiation research*, vol. 154, n° 2, p. 178–186. [12](#)
- PIMBLOTT, S. M. et J. A. LAVERNE. 1992, «Scavenger Concentration Depenences of Yields in Radiation Chemistry», *J. Phys. Chem*, vol. 96, p. 746–752. [36](#)
- POMPLUN, E. et M. TERRISSOL. 1994, «Low-energy electrons inside active DNA models : a tool to elucidate the radiation action mechanisms», *Radiation and environmental biophysics*, vol. 33, n° 4, p. 279–292. [46](#)

- PSZONA, S., J. KULA et S. MARJANSKA. 2000, «A new method for measuring ion clusters produced by charged particles in nanometre track sections of DNA size», *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 447, n° 3, p. 601–607. [46](#)
- RAO, S., M. HUNTLEY, N. DURAND, E. STAMENOVA, I. BOCHKOV, J. ROBINSON, A. SANBORN, I. MACHOL, A. OMER, E. LANDER et E. AIDEN. 2014, «A 3d Map of the Human Genome at Kilobase Resolution Reveals Principles of Chromatin Looping», *Cell*, vol. 159, n° 7, p. 1665–1680. [41](#), [43](#)
- RAYNAUD, C., A. C. MALLORY, D. LATRASSE, T. JEGU, Q. BRUGGEMAN, M. DELARUE, C. BERGOUNIOUX et M. BENHAMED. 2014, «Chromatin meets the cell cycle», *Journal of Experimental Botany*, vol. 65, n° 10, p. 2677–2689. [41](#)
- ROSSI, H. H. 1959, «Specification of Radiation Quality», *Radiation Research*, vol. 10, p. 522–531. [8](#)
- ROSSI, H. H. 1968, *Microscopic energy distribution in irradiated matter*, f. h. attix and w. c. roesh éd., n° 1 dans Radiation Dosimetry, Academic press New York. [8](#)
- SALVAT, F., J. FERNANDEZ VAREA et J. SempaU. 2009, *PENELOPE 2008 A code system for Monte Carlo simulation of electron and photon transport. Workshop proceedings. Barcelona Spain 30 June 3 July 2008*, universitat de barcelona éd., OECD, Spain, ISBN 978-92-64-99066-1 92-64-99066-6. [15](#)
- SCHRAM, R. D., G. T. BARKEMA et H. SCHIESSEL. 2013, «On the stability of fractal globules», *The Journal of chemical physics*, vol. 138, n° 22, p. 224 901. [43](#)
- SHEPHERD, V. 2006, «The Cytomatrix as a Cooperative System of Macromolecular and Water Networks», dans *Current Topics in Developmental Biology*, vol. 75, Elsevier, ISBN 978-0-12-153175-1, p. 171–223. [38](#)
- SPOTHEIM-MAURIZOT, M., M. MOSTAFAVI, T. DOUKI et J. BELLONI. 2008, *Radiation Chemistry : From basics to applications in material and life science*, EDP Sciences. [32](#)
- STAPLE, D. W. et S. E. BUTCHER. 2005, «Pseudoknots RNA structures with diverse functions», *PLoS Biol*, vol. 3, n° 6, p. e213. [39](#)
- STROM, D. J. 2003, *Health impacts from acute radiation exposure*, Pacific Northwest National Laboratory. [12](#)
- TAKAHASHI, M. 1989, «A fractal model of chromosomes and chromosomal DNA replication», *Journal of theoretical biology*, vol. 141, n° 1, p. 117–136. [41](#)
- TERRISSOL, M., E. POMPLUN et C. MARTIN. 2002, «Computer simulation of 57fe bleomycin Auger effects in DNA», *Radiation protection dosimetry*, vol. 99, n° 1-4, p. 69–72. [23](#)
- THOMSON, R. M. et I. KAWRAKOW. 2011, «On the Monte Carlo simulation of electron transport in the sub-1 keV energy range», *Medical Physics*, vol. 38, n° 8, p. 4531. [48](#)
- TOWNSEND, L. W. 2005, «Implications of the space radiation environment for human exploration in deep space», *Radiation Protection Dosimetry*, vol. 115, n° 1-4, p. 44–50. [12](#)
- TREMETHICK, D. J. 2007, «Higher-Order Structures of Chromatin: The Elusive 30 nm Fiber», *Cell*, vol. 128, n° 4, p. 651–654. [40](#)

VILLAGRASA, C., Z. FRANCIS et S. INCERTI. 2010, «Physical models implemented in the GEANT4-DNA extension of the GEANT-4 toolkit for calculating initial radiation damage at the molecular level», *Radiation Protection Dosimetry*, vol. 143, n° 2-4, p. 214–218. [24](#)

WATSON, J. D. et F. H. C. CRICK. 1953, «Molecular Structure of Nucleic Acids : A Structure for Deoxyribose Nucleic Acid», *Nature*, vol. 171, n° 4356, p. 737–738. [39](#)

WILSON, W. E., J. H. MILLER, D. J. LYNCH, R. R. LEWIS et M. BATDORF. 2004, «Analysis of low-energy electron track structure in liquid water», *Radiation research*, vol. 161, n° 5, p. 591–596. [23](#)

Chapitre 2

Modèle géométrique de l'ADN

« Créer, c'est aussi donner une forme à son destin. »

Albert Camus, Le Mythe de Sisyphe

Sommaire

2.1 Développement de DnaFabric	57
2.1.1 Introduction	57
2.1.2 Architecture du projet	58
2.1.3 Implémentation d'un modèle géométrique dans DnaFabric	61
2.1.4 Visualisation 3D	66
2.2 Modélisation de l'ADN dans DnaFabric	73
2.2.1 De la molécule au nucléosome	73
2.2.2 Structures de haut niveau formées par l'ADN	78
2.3 Modélisation de noyaux cellulaires	81
2.3.1 Modélisation du génome humain	82
2.3.2 Algorithmes de remplissage du noyau	86
2.3.3 Les noyaux cellulaires	89
Références	95

Une simulation Monte-Carlo requiert la création d'une géométrie et d'une cible. La géométrie représente le monde dans lequel la simulation se déroule tandis que la cible définit la zone d'intérêt en fonction des résultats attendus. Dans ce travail de thèse, l'objectif est le calcul des dommages radio-induits précoces dans un noyau cellulaire suite à une irradiation par des ions ou des protons et, dans ce cas, les cibles potentielles sont nombreuses (ADN, membranes, protéines etc.). Les travaux présentés dans ce mémoire se concentrent sur une cible précise : l'ADN contenu dans le noyau cellulaire. Une cible géométrique représentant l'ADN du génome humain, soit environ 6.10^9 paires de nucléotides, est donc réalisée. Ce chapitre est consacré à la présentation du modèle utilisé en tant que cible, ainsi qu'aux outils développés pour le concevoir. Nous commencerons par introduire le logiciel DnaFabric développé pour cela, ensuite la modélisation de l'ADN réalisée dans DnaFabric sera présentée et, enfin, nous décrirons comment la modélisation de l'ADN a été étendue pour prendre en compte l'ensemble du génome humain.

2.1 Développement de DnaFabric

2.1.1 Introduction

La modélisation de l'ADN est un élément primordial des simulations multi-échelles dédiées au calcul des dommages radio-induits précoces. Dans ce cadre, il existe différents modèles de l'ADN avec des complexités variables allant de la simple représentation cylindrique [BUG et al., 2012; GARTY et al., 2010] à des représentations atomistiques très détaillées [BERNAL et al., 2014; FRIEDLAND et al., 2011]. La construction de ces modèles détaillés est complexe, d'une part en raison des nombreux éléments à prendre en compte et, d'autre part, en raison de la structure spatiale intrinsèque de l'ADN. En effet, des travaux en biologie ont montré que la répartition de l'ADN contenu dans un noyau cellulaire est différente selon l'échelle spatiale considérée. Ainsi, si la structure en double hélice de l'ADN est généralement considérée comme bien définie, ce n'est pas le cas des structures formées à de plus hauts niveaux. Ces dernières sont décrites différemment par plusieurs modèles biologiques [BRANCO et POMBO, 2007] et dépendent de la partie du cycle cellulaire considérée. L'existence de ces différentes structures illustre le besoin d'un outil capable de générer des géométries représentatives de l'ADN qui soient variées, multi-échelles et flexibles.

L'informatique graphique est un domaine de recherche dans lequel la manipulation d'objets et de structures en 3D est étudiée en profondeur et dans lequel de nombreuses techniques sont développées à cet effet. Partant de cette constatation, le projet (C++) DnaFabric fut initié pour appliquer ces techniques développées en informatique graphique à la création, manipulation et visualisation de modèles d'ADN complexes. Les géométries créées dans le cadre de ce code sont exportables, notamment vers Geant4, de manière à les utiliser dans des simulations physiques, physico-chimiques et chimiques. DnaFabric est implémenté en C++ et permet de travailler avec les 6.10^9 paires de nucléotides du génome humain. Un code aux objectifs similaires existe [HORNUS et al., 2013] mais il n'est pas dédié à l'implémentation de différents modèles d'ADN exportables vers des codes de simulation de Monte Carlo tels que Geant4. Globalement, DnaFabric se caractérise par des capacités d'édition, de simulation, de sauvegarde et d'export des modèles d'ADN. Enfin, le projet contient son propre moteur de rendu 3D ce qui lui permet d'afficher en temps réel des géométries complexes telles qu'un noyau cellulaire et son génome.

2.1.2 Architecture du projet

Le code source de DnaFabric est divisé en plusieurs modules correspondant aux différentes capacités du logiciel. Cette architecture a été choisie pour faciliter de futures modifications ou extensions tout en séparant les différents domaines d'expertise. Les différents modules du code source de DnaFabric sont les suivants :

Run Prend en charge les processus globaux de DnaFabric tels que l'initialisation, l'arrêt et la boucle principale. Cette dernière s'exécute 60 fois par seconde et génère à chaque fois un affichage 3D.

GUI Définit l'interface graphique globale de DnaFabric. Il s'agit de la fenêtre principale contenant l'espace dédié à l'affichage 3D et des onglets contenant diverses options (styles de caméra, modèle à afficher etc.) et boutons. L'ensemble de l'interface graphique est basé sur la librairie Qt5.

3D Contient l'ensemble des éléments nécessaires à l'affichage en 3D de la géométrie considérée. C'est dans ce module que le moteur de rendu est implémenté. Le moteur de rendu encapsule les fonctions de bas niveau d'OpenGL pour créer un niveau d'abstraction supplémentaire entre l'utilisateur et l'affichage 3D avec OpenGL. Ce niveau d'abstraction facilite grandement l'usage des fonctionnalités 3D de DnaFabric.

Models Module dans lequel se trouve l'implémentation du modèle d'ADN utilisé pour créer les différentes géométries présentées dans ce travail de thèse. Le modèle d'ADN est défini sous la forme de plusieurs objets et routines à assembler pour construire une géométrie.

Engine Implémente des outils pour réaliser des calculs sur les modèles géométriques générés par DnaFabric. Ces calculs sont effectués sur des "threads" ¹ différents du thread principal qui s'occupe exclusivement de l'interface utilisateur et du rendu 3D. Cette parallélisation permet d'effectuer des calculs complexes sans diminuer les performances 3D du moteur de rendu tout en rendant possible la mise à jour de l'affichage en fonction de l'avancé des calculs.

Utility Rassemble diverses méthodes pouvant être appelées n'importe où dans le code.

External Contient les librairies et codes extérieurs au projet mais néanmoins utilisés.

Il est important d'insister dès à présent sur la distinction entre un objet géométrique, un modèle géométrique et une géométrie. La géométrie constitue l'ensemble du monde décrit par l'utilisateur tandis qu'un objet géométrique en est un unique élément. Un modèle géométrique désigne un prototype d'objet géométrique représentatif d'un élément du monde réel tel qu'une protéine ou une molécule. Les objets géométriques sont créés à partir de ces prototypes. En informatique, un modèle géométrique correspond à une classe tandis qu'un objet géométrique à une de ces instances. En effet, une classe est un modèle à partir duquel sont créées des instances correspondant à des objets. Par exemple, considérons une géométrie constituée d'une protéine elle-même formée de 10 molécules identiques définies dans DnaFabric. Dans cet exemple, il y a 11 objets géométriques (la protéine ainsi que chacune des molécules) et seulement deux modèles géométriques (protéine et molécule). Toujours dans cet exemple, le modèle géométrique de la protéine caractérise l'ensemble de la géométrie. Ce qui n'est plus le cas si la géométrie contient deux protéines au lieu d'une seule.

1. Un thread correspond à une entité capable d'exécuter une série d'instructions de manière indépendante.

La figure 2.1 décrit l'ensemble des processus s'exécutant suite au démarrage de DnaFabric. Pour commencer, le programme cherche et liste dans un tableau toutes les géométries ADN disponibles. La première entrée de ce tableau constitue la géométrie qui sera affichée par défaut. Ensuite, le programme initialise l'interface graphique puis le contexte OpenGL grâce à la librairie Qt5. Une fois le contexte OpenGL en place, le programme débute les itérations sur sa boucle principale et effectue 60 itérations par seconde. Cela correspond à une fréquence de rafraîchissement de 60 images par seconde pour l'affichage 3D. Une telle fréquence de rafraîchissement est indiscernable pour l'œil humain et crée l'illusion d'un affichage 3D fluide. Lors de chaque itération de la boucle principale, plusieurs actions sont effectuées. D'abord, le programme vérifie si la géométrie sélectionnée a déjà été chargée en mémoire. Si ce n'est pas le cas alors le programme lance son chargement. Lors de la première itération de la boucle principale, il n'y a pas de géométrie précédemment chargée et le processus de chargement de la géométrie "par défaut" débute donc systématiquement. Le chargement d'une géométrie est constitué de deux étapes : premièrement, toutes les positions, orientations et formes des objets 3D sont importées dans la mémoire vive également nommée "Random Access Memory" (RAM) puis, dans un second temps, elles sont transférées de la RAM vers la mémoire de la carte graphique. Suite à cette étape de vérification et éventuel chargement, l'affichage de la géométrie peut véritablement commencer. Il existe deux modes pour générer cet affichage : "animated" ou "not animated". "Not animated" signifie que le contenu de la mémoire de la carte graphique (ou "mémoire vidéo") sera directement affiché sur l'écran tandis qu'"animated" implique que la mémoire vidéo sera mise à jour avant de procéder à l'affichage. Cette mise à jour consiste à prendre en compte tous les éventuels changements effectués sur la géométrie suite à l'itération précédente et à les transposer successivement dans la RAM puis dans la mémoire vidéo. L'affichage à partir de la mémoire vidéo s'effectue au travers de fonctions OpenGL appelées dans un ordre précis. Suite à l'affichage, les éventuelles actions sur les éléments de l'IG sont prises en compte. Enfin, lorsque toutes ces étapes sont effectuées alors le programme se met brièvement en veille avant de débiter une nouvelle itération de la boucle principale recommençant ainsi les étapes précédentes depuis la vérification du statut de la géométrie sélectionnée. La durée de cette mise en veille est variable et permet d'avoir précisément 60 itérations par seconde tout en minimisant la charge de travail du thread principal.

Le projet DnaFabric est utilisable de deux façons : en tant que simple exécutable ou en tant que librairie. L'exécutable est aisé à mettre en œuvre mais l'utilisateur ne peut alors qu'utiliser ce qui a déjà été codé dans le projet. A l'inverse, l'utilisation de DnaFabric en tant que librairie est plus complexe mais permet à l'utilisateur d'ajouter de nouveaux éléments. L'utilisation du projet en tant que librairie est recommandée pour cette raison et requiert la création d'une application utilisateur. Cette application doit contenir, au minimum, les éléments suivants :

- Un fichier *CMakeLists.txt* utilisé pour identifier la librairie DnaFabric installée chez l'utilisateur, compiler l'application et générer le fichier exécutable à partir des fichiers sources.
- Un fichier source *main.cc* constituant la base de l'application. Ce fichier active et initialise DnaFabric tout en y injectant les fichiers de l'application utilisateur.
- La classe *UserApp* (*UserApp.cc* et *UserApp.hh*) qui établit une configuration pour l'ensemble des réglages de DnaFabric. Cette classe introduit également la classe *World* mentionnée ci-après.
- La classe *World* dans laquelle l'utilisateur définit une géométrie à partir de différents objets géométriques.

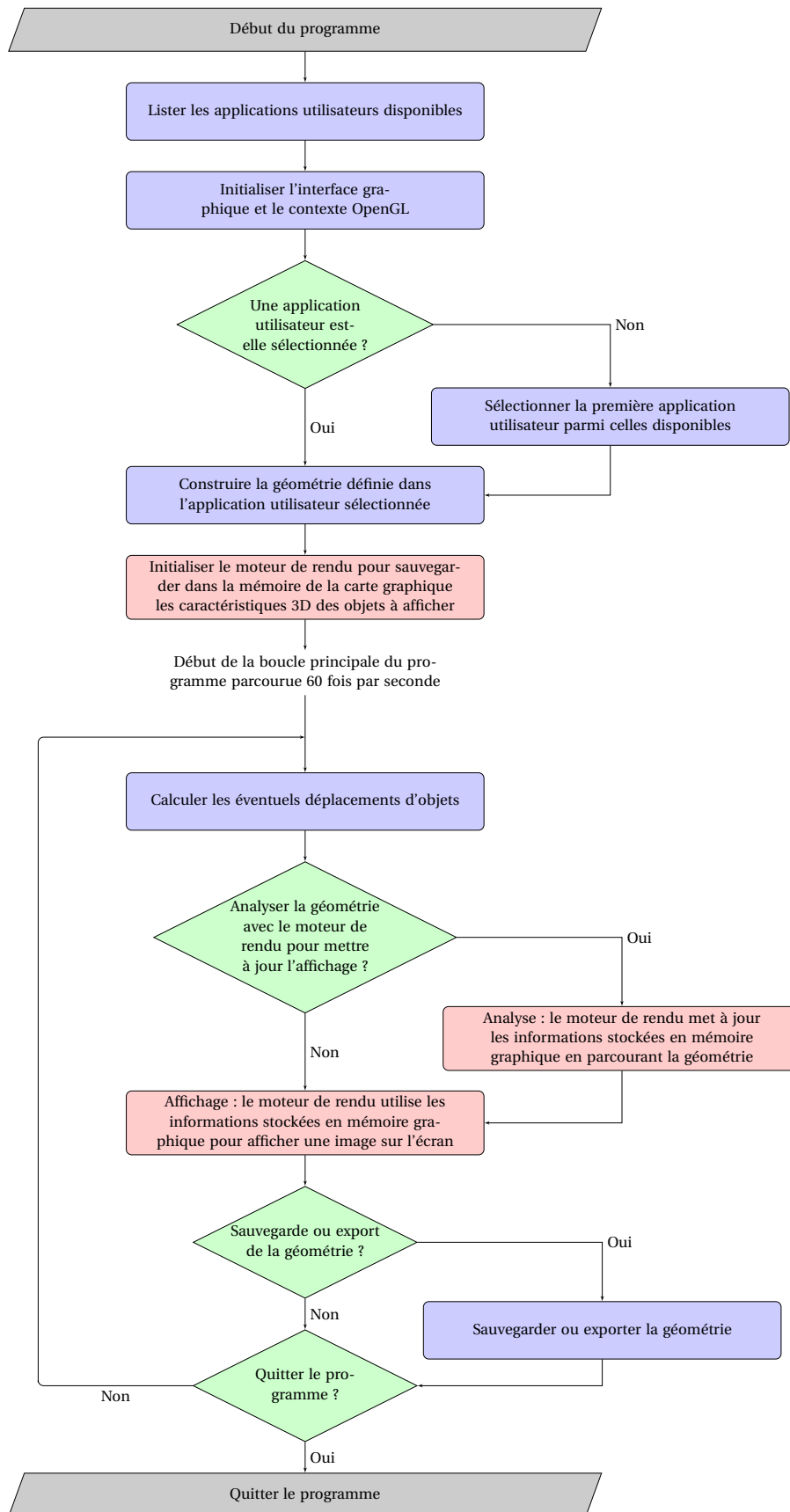


FIGURE 2.1 – Logigramme simplifié de DnaFabric.

L'application est un programme en soi qui fait appel aux différentes fonctions de DnaFabric et qui génère un exécutable lors de la compilation de son code source. Chaque application doit définir une géométrie en créant des objets à partir de modèles géométriques pour les insérer dans la classe *World* qui représente le volume mère de toute la géométrie définie par l'utilisateur. L'ajout d'objets dans la classe *World* n'est donc possible qu'à condition d'avoir à disposition des modèles géométriques. Ces derniers peuvent être définis par défaut dans le code de DnaFabric ou définis directement dans l'application utilisateur. Dans ce dernier cas, l'utilisateur doit créer une nouvelle classe (fichiers .cc et .hh) et l'inclure dans le fichier *CMakeLists.txt*. L'utilisateur définit ensuite le modèle géométrique directement dans la classe ainsi créée.

2.1.3 Implémentation d'un modèle géométrique dans DnaFabric

Deux approches pour construire un modèle géométrique

Il existe plusieurs approches pour construire un modèle géométrique et la plus commune est l'approche "équation" qui consiste à générer dans un premier temps toutes les positions et orientations des objets du modèle géométrique au moyen d'équations pour, dans un second temps, les appliquer aux objets. L'approche "objet" est une approche dans laquelle les objets du modèle géométrique sont positionnés et orientés au fur et à mesure de la construction grâce à l'utilisation de différents paramètres clefs. DnaFabric rend possible la mise en œuvre de ces deux approches lors de la construction d'un modèle géométrique. Cependant, l'approche dite "objet" est préférée en raison de son ergonomie et de sa simplicité. A titre d'exemple, la figure 2.2 contient 13 pavés disposés le long d'un chemin décrit par deux segments s_1 et s_2 définis par les équations 2.1 et 2.2. Les centres des pavés sont séparés deux à deux par une même distance d .

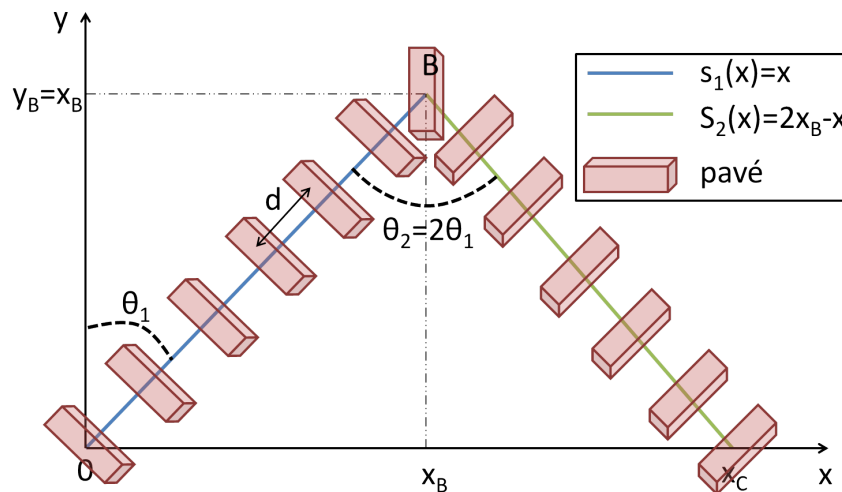


FIGURE 2.2 – Placement de 13 pavés le long de deux segments s_1 (bleu) et s_2 (vert). Les pavés sont séparés deux à deux d'une distance d .

$$\forall x \in [0, x_B] : s_1(x) = x \quad (2.1)$$

$$\forall x \in [x_B, x_C] : s_2(x) = 2 \times x_B - x \quad (2.2)$$

La description de cette géométrie 2D est effectuée par les algorithmes 2.1 et 2.2 qui utilisent respectivement l'approche "équation" et l'approche "objet". Avec l'approche "équation", les éléments nécessaires à la construction sont :

- Les équations 2.1 et 2.2.
- La distance d .
- Les angles θ_1 et θ_2 .

Dans le cas de l'approche "objet", les éléments nécessaires sont les suivants :

- La distance d .
- Les angles θ_1 et θ_2 .
- Les valeurs x_B et y_B .

L'approche "objet" est considérée comme plus simple car elle requière moins d'informations initiales et son algorithme (cf. algorithme 2.2) est le plus court et le plus facilement adaptable. Par exemple et dans cette approche, une augmentation de la distance d uniquement entre les pavés 3 et 4 décalera mécaniquement les pavés suivants positionnés après-coup. Ainsi, cette modification conserve la cohérence de l'ensemble de la structure, c'est à dire les écarts et orientations entre les pavés pris deux à deux. A l'inverse et dans le cas de l'approche "équation", cette modification ne serait pas répercutée immédiatement sur l'ensemble des pavés suivants et introduirait une incohérence rendant nécessaire l'application d'une correction. Ceci est particulièrement significatif dans le cas de modèles complexes tels que les modèles géométriques représentatifs de l'ADN où les corrections à appliquer deviennent rapidement complexes. Le projet DnaFabric contient divers outils facilitant l'écriture du code descriptif d'un modèle géométrique avec les deux approches : repère local, changement de repère, translation, rotation etc. Cependant, seule l'approche "objet" en fait un usage complet.

Organisation des objets dans un modèle géométrique

Il est possible de mettre en évidence des relations d'inclusion qui lient les différents objets présents dans les modèles géométriques représentant l'ADN. Par exemple, *"une molécule contient des atomes"*. L'utilisation de ces relations d'inclusion lors de la création d'un modèle permet d'organiser efficacement les objets géométriques les uns par rapport aux autres tant du point de vue du code informatique que des calculs effectués. La première implémentation envisagée pour ces inclusions hiérarchiques est généralement l'**arbre géométrique** illustré en figure 2.3a. Cette implémentation est toutefois limitée car un objet enfant ne peut avoir qu'un objet parent. A contrario, l'implémentation d'un **graphe** tel que représenté en figure 2.3b ne présente pas cette limitation. La capacité d'associer plusieurs parents à un objet enfant offre un avantage considérable en permettant de créer de légers objets-façades qui remplacent de lourds objets. Ainsi, la quantité de mémoire RAM dédiée à la gestion de plusieurs objets complexes peut être significativement réduite. La figure 2.4 illustre cet avantage à l'aide d'un exemple constitué de trois objets enfants complexes (points bleus) à inclure dans un objet parent (point rouge). En figure 2.3a, un arbre est utilisé et trois objets enfants complexes sont créés en mémoire. L'objet parent devient alors très volumineux. Dans le graphe de la figure 2.4b, un seul objet complexe est instancié ainsi que trois objets-façades qui y font référence au moyen de relations parent/enfant : chaque objet-façade est un parent distinct de l'unique objet complexe. Du point de vue d'un utilisateur, les trois objets-façades sont perçus comme trois copies de l'objet complexe mais, du point de vue de la mémoire, il ne font que *représenter* un seul et unique objet complexe. La majorité de la mémoire RAM utilisée par l'objet parent de la figure 2.4b provient alors de l'unique objet complexe à stocker, les objets-façades étant particulièrement légers. Au final, l'objet parent de la figure 2.4b est plus léger que l'objet parent de la figure 2.4a.

Données : Les segments s_1 et s_2 , les paramètres $\theta_1, \theta_2, x_B, y_B, d$ et le modèle géométrique d'un pavé
Résultat : 13 pavés sont positionnés et orientés selon le chemin tracé par les deux segments s_1 et s_2 .

Initialisation.

Calculer d_x qui a la même valeur pour s_1 et s_2 avec : $d_x = d \times \cos \theta_1$

$num_{pavé} = 0$

pour tous les 7 pavés du segment s_1 faire

Calculer la position $P_{num_{pavé}}$ avec :

$$P_{num_{pavé}} = \begin{cases} x_{pavé} = d_x \times num_{pavé} \\ y_{pavé} = s_1(x) = d_x \times num_{pavé} \end{cases}$$

Sauvegarder la position $P_{num_{pavé}}$.

$num_{pavé} = num_{pavé} + 1$

fin

pour tous les 6 pavés du segment s_2 faire

Calculer la position $P_{num_{pavé}}$ avec :

$$P_{num_{pavé}} = \begin{cases} x_{pavé} = d_x \times num_{pavé} \\ y_{pavé} = s_2(x) = 2 \times x_B - d_x \times num_{pavé} \end{cases}$$

Sauvegarder la position $P_{num_{pavé}}$.

$num_{pavé} = num_{pavé} + 1$

fin

Définir une matrice de rotation 2D :

$$R_{num_{pavé}}(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

$num_{pavé} = 0$

pour tous les 6 premiers pavés du segment s_1 faire

Sauvegarder la matrice de rotation du pavé $R_{num_{pavé}}(\theta_1)$.

$num_{pavé} = num_{pavé} + 1$

fin

Sauvegarder la matrice de rotation du pavé 7, soit $R_{num_{pavé}}(2\theta_1)$.

$num_{pavé} = num_{pavé} + 1$

pour tous les 6 pavés du segment s_2 faire

Sauvegarder la matrice de rotation du pavé, soit : $R_{num_{pavé}}(3\theta_1)$.

$num_{pavé} = num_{pavé} + 1$

fin

$num_{pavé} = 0$

pour tous les 13 pavés à placer faire

Générer un objet *obj* à partir du modèle géométrique d'un pavé. L'objet est positionné à l'origine et orienté selon l'axe Y par défaut.

obj.Translation($P_{num_{pavé}}$)

obj.AppliquerMatrice($R_{num_{pavé}}$)

Sauvegarder *obj*.

$num_{pavé} = num_{pavé} + 1$

fin

Données : Les segments s_1 et s_2 , les paramètres θ_1 , θ_2 , x_B , y_B , d et le modèle géométrique d'un pavé

Résultat : 13 pavés sont positionnés et orientés selon le chemin tracé par les deux segments s_1 et s_2 .

Initialisation.
 Créer un objet nommé *pinceau* à partir du modèle géométrique d'un pavé. L'objet *pinceau* est en charge du placement des pavés. Il est positionné à l'origine et orienté selon l'axe Y.
 Orienter l'objet *pinceau* selon l'axe du segment s_1 : *pinceau.Rotation*(θ_1)
pour tous les 6 premiers pavés du segment s_1 faire
 Traduire l'objet *pinceau* : *pinceau.Translation*(d)
 Créer une copie de *pinceau* (même position et orientation) et placer cette copie dans la géométrie.
fin
 Traduire et pivoter l'objet *pinceau* actuellement sur le pavé 6 pour qu'il soit sur le pavé 7 : *pinceau.Translation*(d) et *pinceau.Rotation*(θ_1)
 Pivoter l'objet *pinceau* pour l'orienter selon le segment s_2 : *pinceau.Rotation*(θ_1)
pour tous les 6 pavés du segment s_2 faire
 Traduire l'objet *pinceau* : *pinceau.Translation*(d)
 Créer une copie de *pinceau* (même position et orientation) et placer cette copie dans la géométrie.
fin

ALGORITHME 2.2 – Construction d'une géométrie avec l'approche "objet".

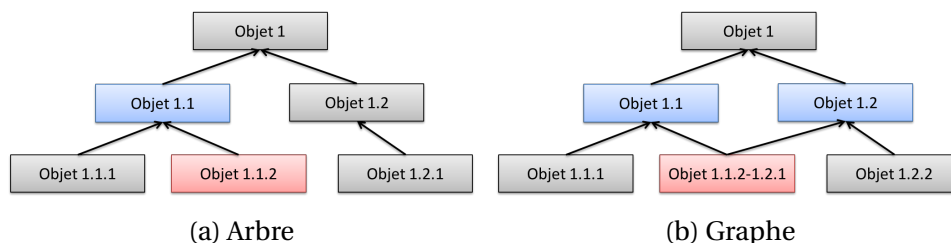
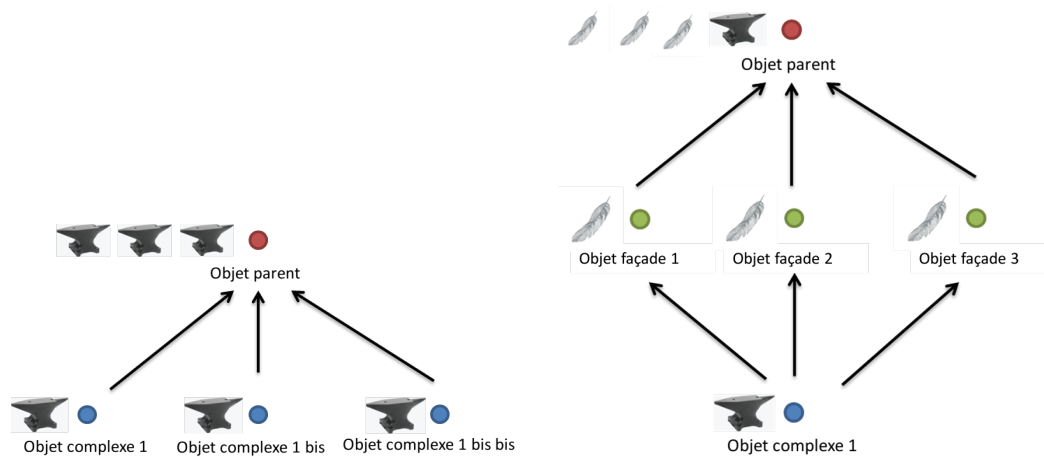


FIGURE 2.3 – Deux organisations hiérarchiques d'objets : l'arbre et le graphe. Dans un arbre chaque enfant ne peut avoir qu'un seul parent tandis que dans un graphe un enfant peut avoir plusieurs parents. En rouge, un objet enfant et, en bleu, son ou ses parent(s).



(a) Empreinte mémoire d'un arbre. Chaque objet complexe possède sa propre instance et son propre espace mémoire.

(b) Empreinte mémoire d'un graphe. Chaque objet façade renvoie vers le même objet complexe.

FIGURE 2.4 – Empreintes mémoires occasionnées par une organisation hiérarchique en arbre ou en graphe. Les deux organisations représentent le même modèle géométrique mais l'organisation en graphe a une empreinte mémoire plus faible.

Calcul en parallèle

Lorsque les géométries définies par l'utilisateur contiennent de nombreux objets le thread principal qui est responsable de l'affichage en 3D se dédie entièrement à cela et d'éventuelles simulations ou calculs ne peuvent y être effectués sans impacter directement les performances 3D. Une solution est de les prendre en charge dans un thread distinct (thread secondaire) du thread chargé de l'affichage (thread principal). Le thread secondaire duplique l'ensemble du modèle géométrique et effectue les calculs et modifications requises sur cette copie. Ensuite, le thread secondaire signale au thread principal la fin des opérations demandées pour enclencher la mise à jour du modèle géométrique initial. Effectuer des tâches en parallèle démultiplie les capacités de calcul de DnaFabric tout en offrant la possibilité de contrôler directement sur l'écran le déroulement d'une simulation.

Sauvegarde et export des modèles géométriques

La création d'un modèle géométrique donné peut se faire directement en C++ dans un fichier dédié, résulter d'une simulation ou encore des actions de l'utilisateur dans le cadre d'un modèle dynamique. Dans les deux derniers cas, le modèle géométrique doit être sauvegardé pour assurer sa reproductibilité. En informatique, l'implémentation d'un mécanisme de sauvegarde revient à mettre en place un système de **sérialisation**. Dans DnaFabric, tous les modèles géométriques directement disponibles dans le code contiennent un tel mécanisme pour sauvegarder tout modèle géométrique qui en ferait usage. Dans le cas où l'utilisateur crée de nouveaux modèles géométriques alors il doit lui-même implémenter les fonctions et paramètres nécessaires à la sérialisation de ce modèle. Concrètement, il s'agit d'ajouter au modèle une fonction *DescribeYourself()*, dont le but est de décrire dans un fichier les valeurs des variables liées au modèle, ainsi qu'une fonction *LoadYourself()* chargée de modifier les variables du modèle en fonction d'un fichier d'entrée. Finalement, lors de la sauvegarde d'un modèle géométrique dans DnaFabric, les informations décrivant le modèle sont écrites dans un fichier texte ".fab" qui, ensuite, peut-être lu pour

reconstruire le modèle géométrique.

Chaque modèle géométrique peut être exporté dans un fichier texte ayant pour extension ".fab2g4dna" indépendamment du processus de sérialisation. Ce fichier d'export est conçu pour être importé dans une application utilisateur Geant4 dédiée. Cette application Geant4 convertit les données géométriques contenues dans le fichier ".fab2g4dna" en volumes utilisables dans une simulation Monte-Carlo effectuée avec Geant4. Un fichier ".fab2g4dna" typique possède la structure type suivante :

1. Généralités au sujet de la géométrie décrite dans le fichier : nom, taille, nombre d'éléments etc.
2. Description de l'ensemble des modèles géométriques de base utilisés dans la géométrie : type (molécule, nucléotide, protéine etc.), nom, taille, etc. Dans la version actuelle de DnaFabric, les modèles de base exportables dans ce fichier doivent être sphériques.
3. Position de chaque modèle de base relative au centre de son conteneur. Il s'agit de la partie habituellement la plus longue du fichier : nom, matériau, brin d'ADN, identifiant, x(nm), y (nm), z (nm).

Chaque ligne du fichier a un but spécifique défini par un premier "mot-balise". Les balises actuellement définies sont listées dans le tableau 2.1. C'est grâce à la lecture du fichier ligne par ligne et à ces balises que l'application construit peu à peu une géométrie dans l'application utilisateur Geant4. Une fois la lecture terminée, le modèle géométrique provenant de DnaFabric est entièrement chargé et utilisable dans Geant4 en tant que géométrie Geant4 typique. Au fur et à mesure de la lecture de ce fichier, l'application Geant4 est capable de détecter d'éventuels chevauchements entre volumes et de les corriger en éditant les volumes concernés. En effet, lorsque l'application utilisateur Geant4 crée une géométrie Geant4 à partir d'un fichier ".fab2g4dna", elle génère, dans Geant4, des sphères pour chaque molécule. Il est possible que deux molécules soient suffisamment proches pour que les sphères les représentant entrent en collision. Dans un tel cas, ces volumes superposés génèrent des chevauchements perturbant le tracking implémenté dans Geant4. Pour éviter de telles perturbations, cet algorithme (cf. chapitre 3) est utilisé pour "découper" les volumes superposés et éviter ainsi tout chevauchement. L'algorithme est codé dans l'application Geant4 utilisée pour la simulation.

TABLEAU 2.1 – Balises utilisées dans le fichier ".fab2g4dna"

"Flag"	Type	Signification
_Name	Information générale	Nom de la géométrie décrite
_Size	Information générale	Taille du volume mère (cubique)
_Number	Information générale	Nombre de volumes dans la géométrie actuelle
_Radius	Forme	Définie un type de volume à placer
_pl	Position	Place un volume à la position spécifiée

2.1.4 Visualisation 3D

OpenGL (Open Graphics Library) est une bibliothèque d'environ 250 fonctions standards écrites en C et dédiées à la génération d'images à partir de **primitives géométriques**. Ces dernières sont des objets 3D construits et définis par l'utilisateur à l'aide des fonctions fournies par OpenGL qui permettent essentiellement de créer et manipuler des triangles. Les fonctions d'OpenGL sont utilisables directement dans un code écrit en C ou

C++ mais peuvent également être utilisées au travers de bibliothèques intermédiaires. Le projet DnaFabric utilise directement les fonctions OpenGL dans le module de visualisation et définit ainsi ses propres primitives géométriques (objets 3D).

Plusieurs primitives géométriques simples sont implémentées dans DnaFabric : sphère, cube, cylindre, ellipse, ellipsoïde, brique. Chaque primitive peut être associée à un objet géométrique de manière à constituer sa représentation en 3D. Une primitive est constituée d'un maillage appelé **mesh** décrivant l'enveloppe surfacique de l'objet 3D. Un mesh est subdivisé en triangles eux-mêmes définis par les coordonnées de leurs sommets appelés **vertex** (**vertices** au pluriel) comme illustré en figure 2.5. D'autres éléments peuvent être associés à une primitive géométrique en plus du mesh :

- Une **couleur** peut être associée à chaque sommet et la couleur d'un triangle est interpolée à partir des couleurs de ses trois sommets.
- Les **textures** sont des images 2D liées à un mesh qui sont projetées sur celui-ci par OpenGL lors de l'affichage. Cette projection, qui consiste à plaquer une image 2D sur un maillage en 3D requière, la définition d'un tableau de correspondance associant à chaque point d'un mesh un point de la texture. Les textures peuvent être utilisées pour associer à un objet 3D des données diverses relatives à son rendu en 3D : ombres, transparence, lumière etc.
- Les **buffers** sont des tableaux stockés en mémoire et utilisés pour enregistrer des informations variées dans le but d'optimiser l'utilisation des fonctions OpenGL responsables de l'affichage d'un objet 3D. Une primitive géométrique peut être liée à des buffers pour optimiser son affichage.

La figure 2.6a représente le rendu 3D d'une primitive géométrique uniquement associée à un mesh. L'ajout d'une couleur rouge pour chaque sommet du mesh produit le rendu reproduit en figure 2.6b tandis que l'ajout d'une texture génère celui de la figure 2.6c. Le relief particulièrement visible sur la figure 2.6b est dû à la juxtaposition de zones claires et sombres simulant respectivement des intensités lumineuses fortes et faibles. D'un point de vue informatique, une zone claire est une zone où les couleurs tendent vers le blanc tandis qu'une zone sombre correspond à des couleurs tendant vers le noir. Le cerveau humain interprète ces variations d'intensités lumineuses pour déterminer le relief d'un objet. Ainsi, une sphère dont la surface est représentée avec exactement la même couleur sera perçue comme un disque tandis qu'avec l'ajout de zones sombres et claires cette sphère sera identifiée en tant que telle par un cerveau humain. En pratique, le rendu de ces zones claires et sombres est directement calculé sur la carte graphique ("Graphics Processing Unit" : GPU) grâce aux instructions contenues dans les **shaders**.

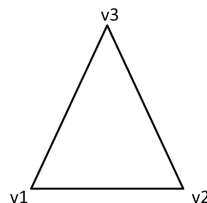


FIGURE 2.5 – Un triangle et ses trois vertices ou sommets (v1, v2 et v3).

Les shaders sont des programmes écrits en GLSL ("OpenGL Shading Language") et exécutés par le GPU juste avant l'affichage sur l'écran d'une primitive géométrique. Le rôle des shaders est de générer l'image 2D à afficher sur l'écran. Cette image est calculée en projetant les primitives géométriques définies en 3D dans un espace 2D. OpenGL utilise plusieurs types de shaders dont les plus courants sont les suivants :

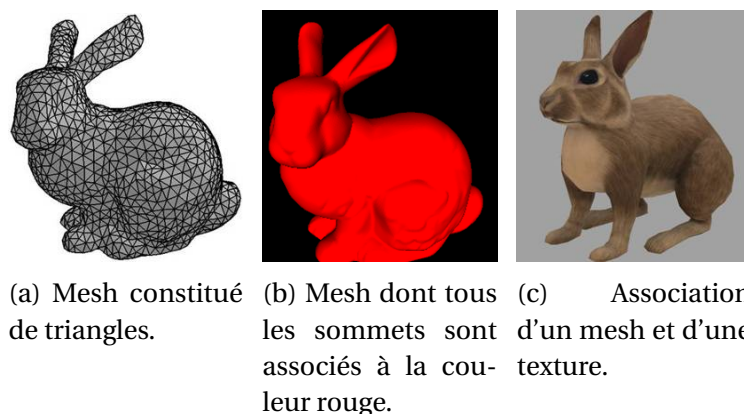


FIGURE 2.6 – L’affichage d’une primitive géométrique selon les informations associées à son mesh.

- Le **vertex shader** prend en charge le calcul de la projection des coordonnées 3D des vertices dans l’espace 2D de l’écran.
- Le **geometry shader** permet la modification du mesh de la primitive géométrique considérée. Ces shaders sont apparus récemment et ne sont pas utilisés dans DnaFabric.
- Le **fragment shader** calcule la couleur de chaque pixel de l’écran en fonction du résultat de la projection préalablement effectuée par le vertex shader.

Les shaders sont exécutés par le GPU dans un ordre précis au sein du **pipeline graphique**. Le pipeline graphique désigne l’ensemble des processus mis en œuvre par OpenGL pour générer une image 2D sur l’écran à partir d’une primitive géométrique en 3D. Une version simplifiée de ce pipeline est présentée en figure 2.7.

La librairie DnaFabric utilise plusieurs techniques d’optimisation associées à des buffers dans le but d’accélérer le processus d’affichage des objet 3D :

- L’**indexing** consiste à retirer de la mémoire les vertices partagés par plusieurs triangles. En effet, deux triangles adjacents ont deux sommets en commun qu’il est inutile de dupliquer pour chaque triangle.
- L’**instancing** est le fait d’afficher plusieurs objets 3D identiques au moyen d’une seule fonction OpenGL contrairement à l’utilisation standard consistant à appeler la fonction OpenGL responsable de l’affichage une fois par objet 3D. Par exemple, l’affichage de 1000 sphères identiques mais à des positions différentes (stockées dans un buffer) ne requiert qu’un seul appel à la fonction d’affichage contrairement au schéma classique qui en nécessiterait 1000.
- L’**union** de plusieurs primitives géométriques pour constituer une super primitive. Cette technique permet d’afficher en une seule fois de nombreux objets 3D différents les uns des autres.
- L’utilisation de **Levels Of Detail (LOD)** [BILJECKI et al., 2014] permet d’associer plusieurs primitives géométriques à un objet de manière à ajuster sa représentation en 3D en fonction de son éloignement de la caméra. Ainsi, la primitive géométrique la plus détaillée sera utilisée pour représenter un objet géométrique proche de la caméra tandis qu’une primitive plus simple sera appelée si l’objet est éloigné. La figure 2.8 explicite ce concept.

L’usage de l’instancing, de l’union et des LODs requiert d’analyser l’ensemble des objets de la géométrie avant chaque affichage. Cette analyse est effectuée par le **moteur de**

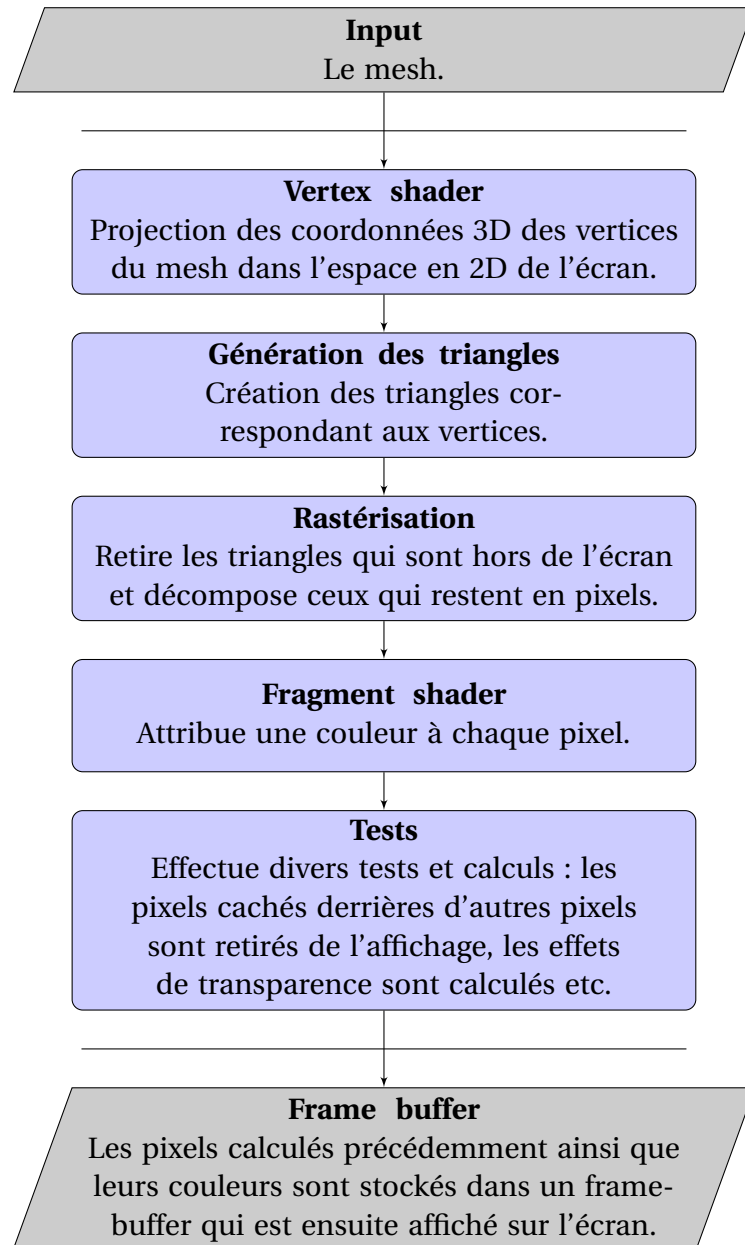


FIGURE 2.7 – Représentation simplifiée du pipeline graphique utilisé par OpenGL.

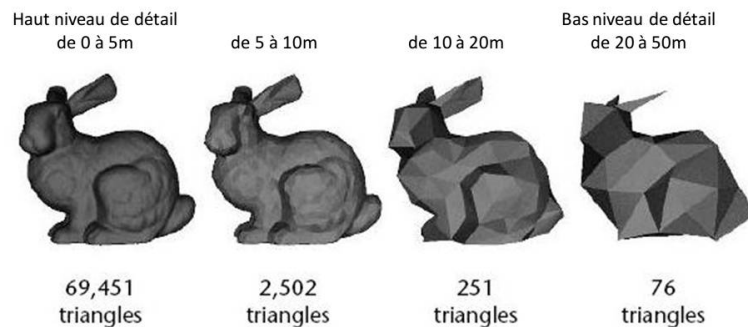


FIGURE 2.8 – Une primitive géométrique et ses quatre niveaux de détails correspondant à quatre meshes différents (source : www.kerry-harnett.blogspot.fr).

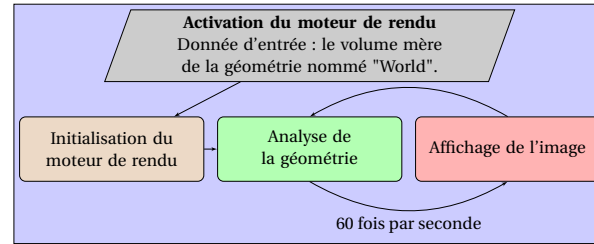
rendu intégré dans le module 3D de DnaFabric et dont le logigramme simplifié est présenté en figure 2.9. Lorsqu'une application DnaFabric débute alors le moteur de rendu s'initialise comme spécifié en figure 2.1. L'initialisation du moteur de rendu consiste à balayer récursivement la géométrie définie par l'utilisateur pour détecter tous les types de primitives géométriques associés aux modèles géométriques et les stocker en mémoire. Dans ce but, le moteur de rendu détermine, pour chaque volume de la géométrie, si une ou plusieurs primitives géométriques lui sont associée(s). Si aucune primitive géométrique n'est associée au volume alors le moteur de rendu passe au volume suivant. Cependant, si au moins une primitive est détectée alors une question se pose : s'agit-il d'une primitive simple ou d'une primitive composite? Dans ce dernier cas, la primitive composite ne contient pas les informations nécessaires à sa représentation 3D mais uniquement la liste des primitives simples qui la compose. En effet, une primitive composite est constituée par l'union de plusieurs primitives simples. Dans une telle situation, le moteur de rendu se charge de créer la primitive composite après avoir préalablement vérifié qu'elle n'est pas déjà présente en mémoire. En effet, le moteur de rendu ne stocke en mémoire qu'un seul exemplaire de chaque primitive simple ou composite. A la fin de l'initialisation, l'espace mémoire dédié au stockage des primitives ne contient donc qu'un seul représentant pour chaque type de primitive géométrique. Le tableau 2.2 schématise la structure de cet espace mémoire et, suite à l'initialisation, ce tableau ne contient que la ligne intitulée "Types de primitive". Une fois l'initialisation terminée, le moteur de rendu intègre le cycle de la boucle principale de DnaFabric où deux tâches lui sont confiées : analyser et afficher la géométrie. L'analyse de la géométrie est similaire à l'initialisation car l'ensemble des volumes de la géométrie sont interrogés récursivement. Cependant, cette fois l'objectif est de compléter le tableau 2.2 avec la position, l'orientation et la couleur de chaque primitive rencontrée. Pour cela, le moteur de rendu détermine si des primitives sont associées à chaque volume de la géométrie et le cas échéant il en choisit une parmi celles disponibles. Le choix dépend de la distance caméra-volume et d'un intervalle spécifié dans les données de chaque primitive. Cet intervalle définit les distances inférieure et supérieure dans lesquelles une primitive donnée est visible comme illustré en figure 2.8. Les données caractéristiques de la primitive (position, orientation et couleur) sont ensuite ajoutées dans une case du tableau 2.2. L'apparition d'une nouvelle case dans une colonne correspondant à un type de primitive signifie qu'un exemplaire de ce type de primitive doit être affiché. Les caractéristiques nécessaires à cet affichage sont contenues dans la case ajoutée. Chaque exemplaire d'un type de primitive à afficher est une **instance** de ce type. Une fois l'analyse effectuée, le moteur de rendu débute sa seconde tâche au sein de la boucle principale : l'affichage. Il s'agit alors d'itérer sur tous les types de primitive présents en mémoire (cf. le tableau 2.2) et de déclencher l'affichage de leurs instances. Pour un type donné, les données des instances associées sont d'abord transférées de la mémoire RAM (CPU) vers des buffers situés dans la mémoire graphique (GPU). Ensuite, une fonction OpenGL particulière est appelée : *glDrawElementsInstanced*. Cette fonction permet d'afficher simultanément plusieurs instances d'un même type si les données requises sont présentes dans des buffers préalablement sélectionnés.

TABLEAU 2.2 – Organisation de l'espace mémoire représentant le tableau des primitives à afficher. Les types de primitives représentés sont le cube, la sphère et la brique. Les données caractéristiques des instances sont présentes sous la forme d'une majuscule suivie d'un numéro. Chaque case contenant une donnée est équivalente à un espace occupé en mémoire RAM (CPU).

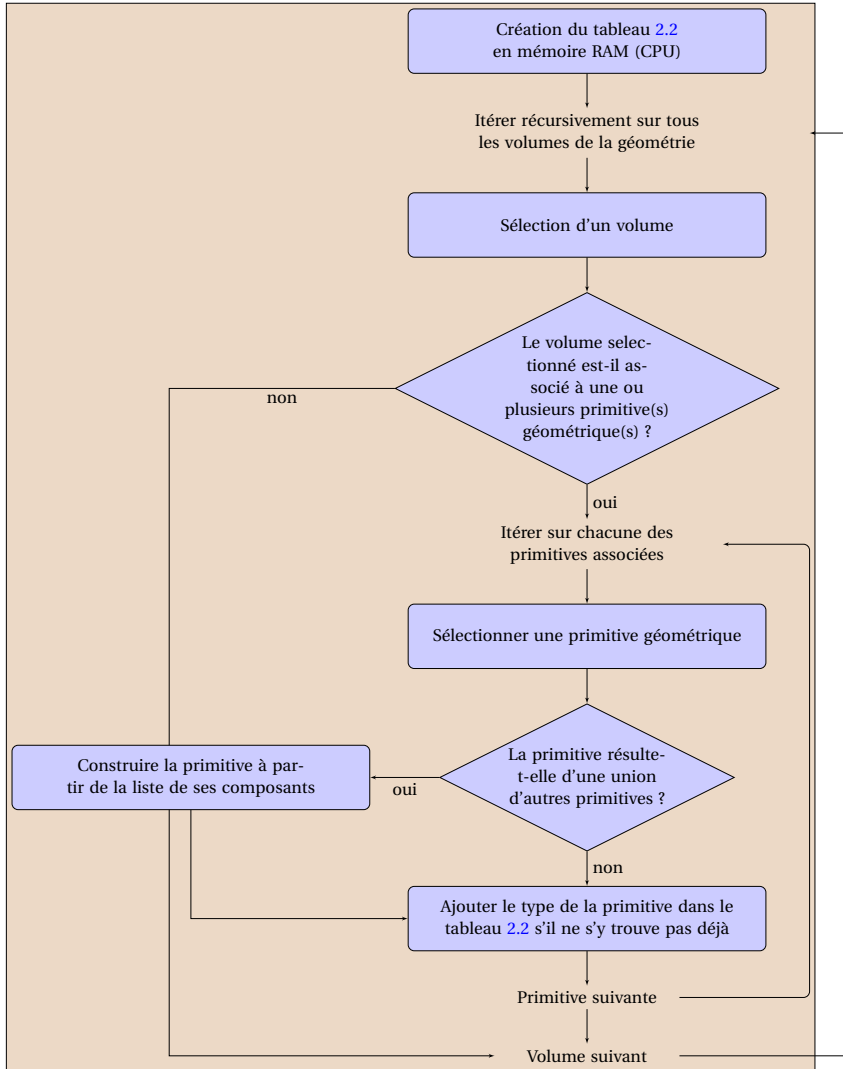
Types de primitive	Cube (C)	Sphère (S)	Brique (B)
Données	C1	S1	B1
	C2	S2	B2
	C3	S3	-
	C4	-	-

Le processus de transfert des positions, orientations et couleur des instances d'un type de primitive de la mémoire RAM (CPU) vers la mémoire graphique (GPU) peut rapidement devenir très chronophage et ralentir le processus d'affichage. Ceci est dû au temps nécessaire pour allouer un nouveau buffer (mémoire graphique) suffisamment grand pour accueillir l'ensemble des données et à un problème de synchronisation lié à l'écriture et la lecture de ce même buffer. En effet, l'allocation d'un nouveau buffer est une opération particulièrement lourde pour OpenGL qui doit avoir lieu le moins souvent possible. Concernant la synchronisation, le point clef est la capacité de la carte graphique à effectuer de nombreuses tâches simultanément comme, par exemple, l'écriture de données dans un buffer et leur lecture. Cependant, la plupart des fonctions d'OpenGL empêchent la carte graphique d'écrire et lire des données simultanément dans un même buffer. Ceci pour éviter une situation où la carte graphique essaierait de lire quelque chose qu'elle n'a pas encore écrit. En pratique cela signifie qu'il faut attendre l'écriture complète des données dans le buffer alloué sur la carte graphique avant de pouvoir déclencher la génération de l'image sur l'écran. Si la quantité de données à écrire est importante alors ce délai peut ralentir significativement le programme. Ainsi, dans le but de minimiser l'allocation de nouveaux buffers sur la carte graphique, un système de transfert de données spécifique a été implémenté dans le moteur de rendu de DnaFabric. Ce système est schématisé en figure 2.10. La mémoire de la carte graphique est divisée en deux catégories dans ce système : les **plages de mémoire** sont de grands buffers capables de contenir plusieurs **zones de mémoire**. Les zones de mémoire ne sont pas des buffers mais des sous-parties de ceux-ci. Chaque zone de mémoire est suffisamment grande pour contenir l'ensemble des informations caractéristiques des instances d'un type de primitive et, donc, peut être utilisée pour effectuer un cycle écriture/lecture correspondant à l'étape "affichage de l'image" sur la figure 2.9 (cf. page 72). Ceci permet donc de minimiser l'allocation de mémoire graphique car plusieurs images peuvent être générées sur l'écran pour une seule allocation. Les zones de mémoire permettent de faire la distinction entre les parties de la plage de mémoire disponibles pour l'écriture ou la lecture d'informations. En effet, le principe consiste à sélectionner une zone de mémoire pour y écrire les informations issues des instances puis, lorsque l'écriture est terminée, déclencher la lecture de cette zone tout en commençant à écrire dans la suivante. Cependant, les zones de mémoire étant dans le même buffer, il est nécessaire d'utiliser une fonction OpenGL particulière *glMapBufferRange* pour lire et écrire simultanément dans ce buffer tout en évitant le mécanisme de sécurité mentionné précédemment. Enfin, lorsque toutes les zones de mémoire d'une plage ont été utilisées alors la plage de mémoire est libérée tandis qu'une nouvelle est allouée. Au final, ce système de découpage combiné à l'utilisation de *glMapBufferRange* permet d'effectuer un transfert de données efficace pour la génération d'images en temps réel.

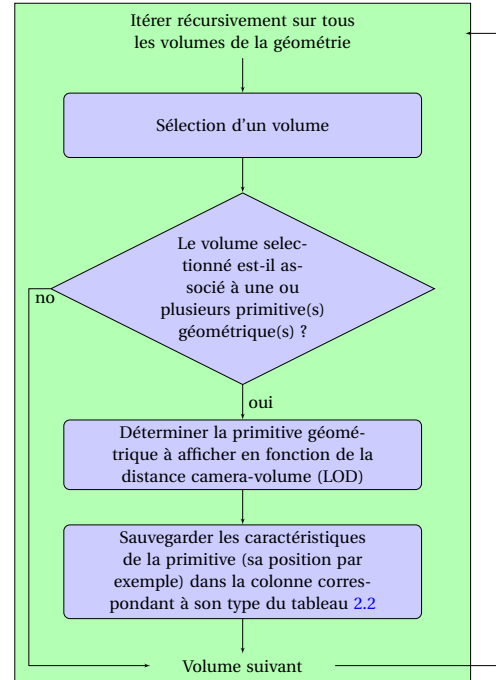
Général



Initialisation



Analyse



Affichage

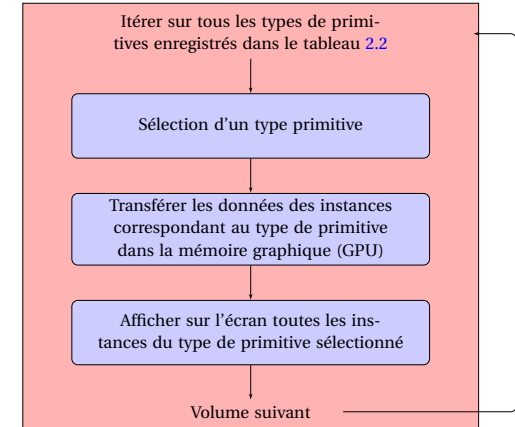


FIGURE 2.9 – Logigramme simplifié du moteur de rendu de DnaFabric.

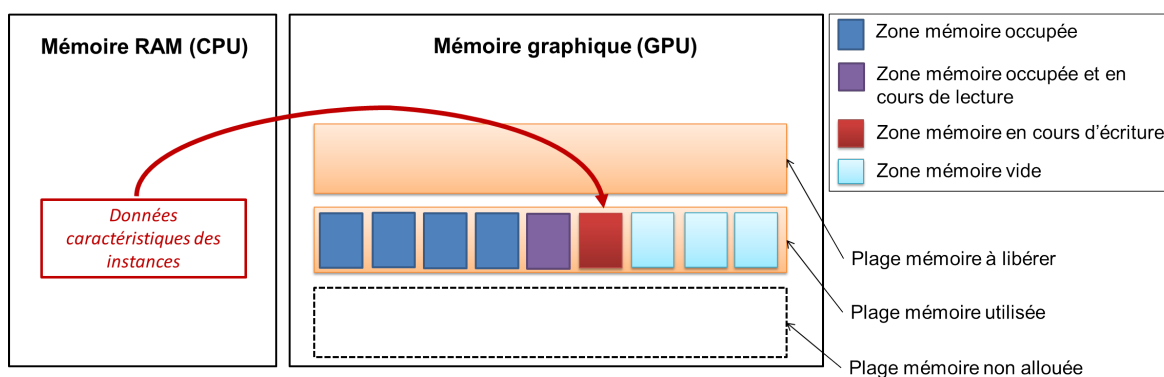


FIGURE 2.10 – Fonctionnement du mécanisme de transfert de données de la mémoire RAM (CPU) vers la mémoire graphique (GPU). Pour rappel, les plages mémoires mentionnées dans ce schéma sont des buffers.

En général et avec un ordinateur moderne (période 2013-2016), l'indexing et l'instancing permettent d'afficher plusieurs dizaines de milliers d'objets simples (des sphères par exemple). L'union de ces objets simples au sein d'entités plus larges permet d'atteindre la centaine de milliers, ce qui représente globalement la limite en puissance des cartes graphiques actuelles. Cependant, l'utilisation des LODs conjointement avec les techniques précédemment mentionnées permet de dépasser cette limite de puissance. Ainsi, dans DnaFabric, il est possible d'afficher en 3D, en temps réel (60 fois par seconde) et de manière globalement satisfaisante, un ensemble de plus de 6 milliards d'objets simples à condition de ne pas les afficher simultanément. Leur sélection pour l'affichage (LODs) doit bien-sûr être intelligemment paramétrée pour obtenir un rendu 3D correct.

2.2 Modélisation de l'ADN dans DnaFabric

Une application utilisateur DnaFabric peut contenir une nouvelle modélisation (ou représentation) de l'ADN ou, à l'inverse, utiliser une modélisation prédéfinie pour construire directement une géométrie de travail. La modélisation prédéfinie de l'ADN est subdivisée en plusieurs composants représentatifs de l'ADN à différentes échelles spatiales. La description de ce modèle est basée sur une hiérarchie détaillée en figure 2.11 où les éléments de base sont les molécules et les éléments de plus haut niveau des fibres ou des voxels. Chaque élément de la figure 2.11 est décrit dans une classe C++ située dans le module "Models" de DnaFabric.

2.2.1 De la molécule au nucléosome

Une paire de nucléotides et sa description moléculaire

Les molécules sont les composants de base du modèle d'ADN développé dans DnaFabric. Chaque nucléotide est constitué de trois molécules (2-désoxyribose, acide phosphorique et une molécule d'adénine, guanine, thymine ou cytosine) représentées par des sphères. Ainsi, une paire de nucléotides est représentée par un assemblage de six sphères car elle contient 6 molécules. Les fichiers PDB issus du projet Glactone [BERNAL et al., 2013] sont utilisés pour déterminer les tailles et positions de chacune de ces sphères. Ces fichiers décrivent atome par atome les paires de nucléotides adénine-thymine et guanine-cytosine. À partir de ces données, le volume de chaque molécule est calculé en sommant les volumes des sphères atomiques (position et rayon de Van Der Waals des atomes) et en

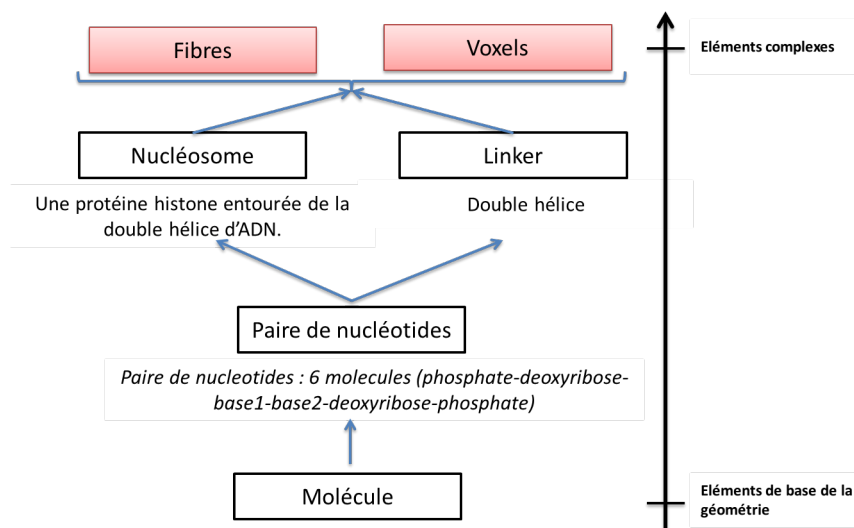


FIGURE 2.11 – Structure hiérarchique du modèle ADN développé dans DnaFabric

soustrayant ceux résultant des chevauchement entre atomes. La position de chaque molécule est déterminée par le barycentre de l'ensemble des atomes la constituant. Les coordonnées du barycentre sont calculées à partir des coordonnées de chaque atome pondérées par son rayon de Van Der Waals. Au final, un volume et une position sont calculés par molécule. Un volume moyen est calculé à partir des volumes des molécules adénine, thymine, guanine et cytosine pour les représenter avec une unique sphère. Le tableau 2.3 indique les volumes obtenus. Chaque paire de nucléotides est placée dans un objet parent

TABEAU 2.3 – Volumes des sphères représentant les molécules des paires de nucléotides adénine-thymine et guanine-cytosine

Type	Volume moyen (nm^3)
2-Désoxyribose	0.091
Acide phosphorique	0.060
Adénine, guanine, cytosine et thymine	0.093

de forme similaire qui modélise une partie de la couche d'hydratation entourant chaque paire de nucléotide. Dans ce travail, une couche d'hydratation d'un volume correspondant à 24 molécules d'eau par paire de nucléotides est prise en compte [FRIEDLAND et al., 2011]. Cette couche d'hydratation permet de détecter d'éventuelles ionisations ou excitations produites par une particule interagissant avec les molécules d'eau la constituant car ces interactions peuvent endommager une molécule d'ADN proche [ALIZADEH et al., 2013]. Au final, une paire de nucléotides contient 6 molécules et une couche d'hydratation, le tout formant un objet géométrique à part entière dans DnaFabric. Cet objet est associé à une primitive géométrique composite générée par le moteur de rendu à partir des molécules qui la composent. Le processus de construction d'une paire de nucléotides est résumé en figure 2.12.

Double hélice de l'ADN

Différents types de double hélice (cf. figure 2.13) peuvent être rencontrés lors de l'observation d'un filament d'ADN. Seul le type intitulé "B-DNA" en figure 2.13 est pris en compte dans cette modélisation de l'ADN car il est le plus répandu dans des cellules vi-

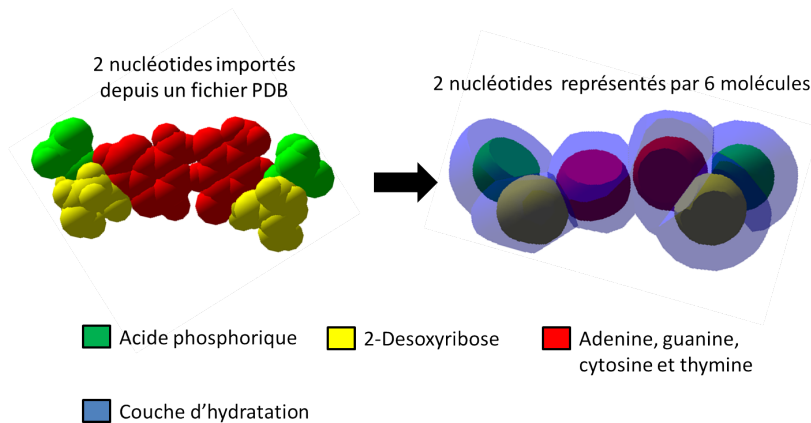


FIGURE 2.12 – La construction d'une paire de nucléotides constituée de molécules à partir des données d'un fichier PDB la décrivant atome par atome.

vantes [LESLIE et al., 1980]. La construction d'une double hélice à partir de paires de nucléotides est aisée dans DnaFabric car chaque paire de nucléotides est un objet géométrique manipulable au travers des outils mentionnés en 2.1.3. Ainsi, il est possible d'empiler les paires de nucléotides et de les faire pivoter de manière à construire une double hélice. Finalement, la construction d'une double hélice de type B à partir de paires de nucléotides se résume à :

1. Se positionner aux coordonnées (0,0,0).
2. Placer la première paire de nucléotides.
3. Effectuer une translation de 0.33 nm le long de l'axe Z [BERNAL et al., 2013].
4. Effectuer une rotation de 36 degrés selon l'axe Z [BERNAL et al., 2013].
5. Placer la seconde paire de nucléotides.

En répétant les étapes de 2 à 5, la double hélice de la figure 2.14 est construite. Cependant cette double hélice est droite, ce qui est rarement le cas en conditions réelles et c'est pourquoi une rotation est souvent ajoutée entre les étapes 1 et 2 de manière à construire une double hélice capable de suivre un chemin quelconque.

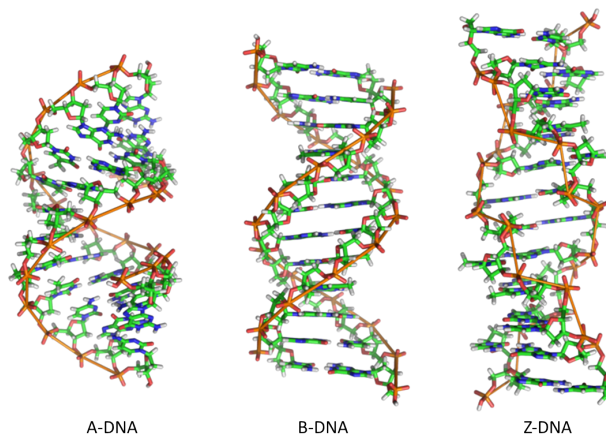


FIGURE 2.13 – Les différents types de double hélice formés par l'ADN (source : wikipedia).

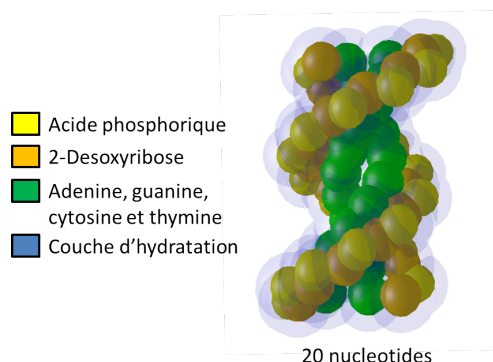


FIGURE 2.14 – Double hélice droite de type B-DNA construite dans DnaFabric.

Nucléosome et linker

Des nucléosomes peuvent être construits et reliés par des linkers en utilisant la double hélice décrite en 2.2.1. Dans ce travail, un nucléosome est un ensemble de protéines histones entouré par un filament d'ADN hélicoïdal de 154 paires de nucléotides. La protéine histone est approximée par une sphère de 2,4 nm de rayon comme le montre la figure 2.15. Le centre de la protéine histone est également le centre de l'objet nucléosome. Les étapes de la construction d'un nucléosome sont les suivantes :

1. Se placer aux coordonnées (0,0,0).
2. Placer la protéine histone.
3. Effectuer une translation du "pinceau" de 4,045 nm selon l'axe X et de -2,370 nm selon l'axe Z.
4. Appliquer une rotation du "pinceau" de -90 degrés autour de l'axe Y.
5. Effectuer une rotation du "pinceau" de 5.33 degrés autour de l'axe X de la paire de nucléotides.
6. Placer la première paire de nucléotides.
7. Exécuter une rotation du "pinceau" de 36 degrés autour de l'axe Z de la paire de nucléotides [BERNAL et al., 2013].
8. Appliquer une rotation du "pinceau" de -4.68 degrés autour de l'axe Y du nucléosome [BERNAL et al., 2013].
9. Effectuer une translation du "pinceau" de 0.33 nm selon l'axe Z de la paire de nucléotides [BERNAL et al., 2013].
10. Placer une paire de nucléotides.

Les étapes de 6 à 9 doivent être répétées 153 fois pour construire un nucléosome avec 154 paires de nucléotides.

Dans une cellule, les nucléosomes sont reliés entre eux par des linkers. DnaFabric est capable de générer de tels linkers en reliant le dernier nucléotide d'un nucléosome donné au premier du nucléosome suivant. Le lien ainsi créé est un enchaînement de paires de nucléotides positionnées et orientées au moyen d'une courbe de Bézier. La géométrie représentée en figure 2.16 est constituée de deux nucléosomes reliés par un tel linker. La génération des linkers est codée dans un outil prenant en charge le positionnement de plusieurs nucléosomes reliés les uns aux autres. Un nombre maximum de paires de nucléotides peut être spécifié pour chaque couple nucléosome-linker et l'outil se chargera d'adapter le linker et le nucléosome. Cette contrainte est fixée à 200 paires de nucléotides par couple nucléosome-linker [KORNBERG, 1974].

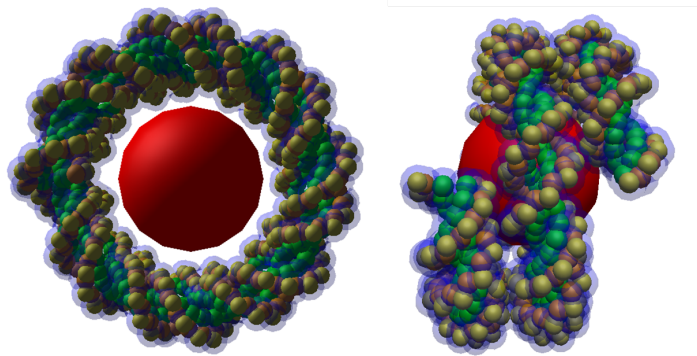


FIGURE 2.15 – Un nucléosome construit dans DnaFabric et contenant une protéine histone centrale (sphère rouge) ainsi que 154 paires de nucléotides.

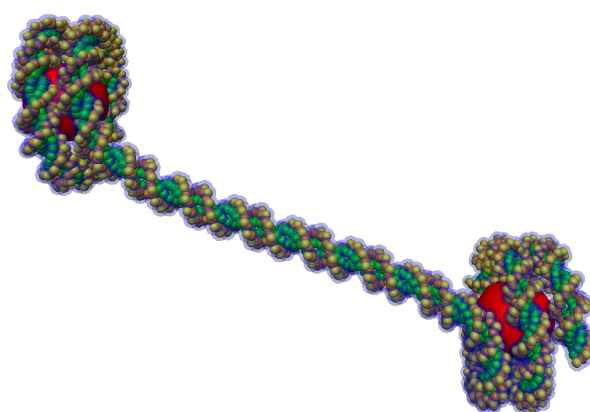


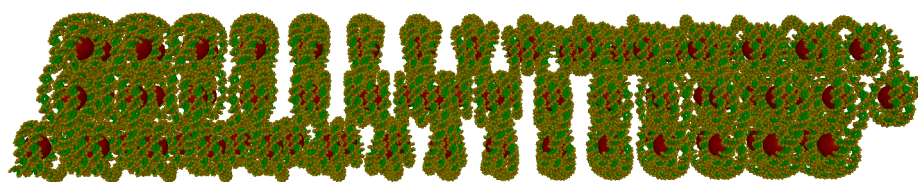
FIGURE 2.16 – Deux nucléosomes reliés par un linker dans DnaFabric.

2.2.2 Structures de haut niveau formées par l'ADN

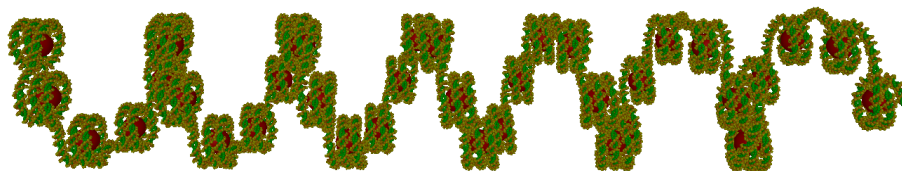
Les éléments constitutifs du modèle ADN décrit en 2.2.1 peuvent être assemblés pour créer des structures de plus haut niveau. Deux catégories de structures de ce type sont présentes par défaut dans DnaFabric : des fibres et des voxels. Chacune de ces structures de haut niveau est également un objet géométrique constitué de divers sous-objets suivant le principe détaillé précédemment (cf. 2.1.3).

Fibres d'hétérochromatine et d'euchromatine

Les fibres sont des agrégats de nucléosomes et linkers disposés hélicoïdalement et typiques du "*lattice model*" [BRANCO et POMBO, 2007]. Deux fibres ont été développées pour approximer deux degrés de compaction de la chromatine dans un noyau cellulaire [DILLON, 2004] : la figure 2.17a représente une chromatine compacte nommée **hétérochromatine** tandis que la figure 2.17b représente une chromatine moins dense appelée **euchromatine**. La fibre d'hétérochromatine possède 130% fois plus de paires de nucléotides que la fibre d'euchromatine [FRIEDLAND et al., 2008] pour le même volume cylindrique. La fibre d'hétérochromatine contient 90 nucléosomes et 18000 paires de nucléotides tandis que la fibre d'euchromatine en contient respectivement 38 et 7600. Les caractéristiques de ces fibres sont indiquées dans le tableau 2.4. La fibre d'euchromatine est directement construite à partir de la fibre d'hétérochromatine dans DnaFabric. Pour cela, 52 nucléosomes sont retirés de la fibre d'hétérochromatine et les nucléosomes restant sont espacés de manière à occuper toute la longueur du cylindre représentant l'enveloppe de la fibre. Les linkers reliant les nucléosomes entre eux s'adaptent automatiquement en s'allongeant, donc en augmentant leur nombre de paires de nucléotides. Cet allongement des linkers déclenche une diminution, également automatique, du nombre de paires de nucléotide appartenant à chaque nucléosome pour conserver 200 paires de nucléotides par couple nucléosome/linker.



(a) Une fibre de chromatine compacte (90 nucléosomes, 18000 paires de nucléotides) représentant l'hétérochromatine.



(b) Une fibre de chromatine moins dense (38 nucléosomes, 7600 paires de nucléotides) représentant l'euchromatine.

FIGURE 2.17 – Fibres d'hétérochromatine et d'euchromatine.

En utilisant les deux modèles de fibres introduits en figure 2.17, une étude a été menée sur l'influence de l'hétéro et euchromatine dans le cadre du calcul des dommages radio-induits précoces à l'ADN [MEYLAN et al., 2016]. Précisons que seule la simulation

de l'étape physique, qui correspond au transport des particules dans la matière, est effectuée dans cette étude. Une première version du code de DnaFabric a également été publiée à cette occasion. Notons cependant qu'il s'agit d'une version limitée qui ne contient aucun des modèles ADN préalablement présentés. En revanche, une grande partie des outils fournis par DnaFabric sont inclus.

TABLEAU 2.4 – Caractéristiques des fibres d'hétérochromatine et d'euchromatine

Caractéristique	Fibre d'hétérochromatine	Fibre d'euchromatine
Nombre de nucléosomes	90	38
Nombre de paires de nucléotides	18000	7600
Pitch	10 nm	42 nm
Rayon de l'hélice	10,46 nm	10,46 nm
Nombre de nucléosomes par tour	6	6
Angle delta	60 deg	60 deg
Rayon de la fibre	17 nm	17 nm
Longueur de la fibre	160 nm	160 nm

Cinq voxels : avant, haut, bas, droite et gauche

Cinq voxels sont implémentés en tant qu'objets géométriques dans le module "Models" de DnaFabric. Chacun de ces voxels est un cube de 50 nm de côté qui contient de l'ADN sous la forme de nucléosomes et linkers hélicoïdalement disposés comme dans le cas de la fibre d'hétérochromatine présentée en figure 2.17a. Pour chaque voxel, le morceau de fibre (nucléosomes et linkers) suit un chemin différent différenciant ainsi les cinq voxels : "avant", "haut", "bas", "droite" et "gauche". L'utilisation conjointe de ces voxels, illustrés en figure 2.18 et décrits par le tableau 2.5, dans un volume parent est un moyen efficace de remplir l'espace avec un **filament d'ADN**. Un tel filament est continu et défini comme un ensemble de voxels consécutifs deux à deux. En effet, les voxels sont conçus pour que deux voxels positionnés côte à côte et correctement orientés forment un filament d'ADN continu comme illustré en figure 2.19. Lorsque deux voxels sont positionnés côte à côte pour former un filament d'ADN alors il se crée une zone de tampon où la densité d'ADN le long du filament est plus faible. La création de cette zone est consécutive à l'impossibilité de maintenir une densité constante sans ajouter des nucléosomes communs aux deux voxels. L'usage de voxels permet de limiter à 5 le nombre d'objets géométriques nécessaires pour la création d'un filament d'ADN ce qui diminue significativement la mémoire requise pour la construction de filaments d'ADN tout en améliorant les performances générales de la simulation (cf. 2.1.3 et la figure 2.4).

Le tableau 2.5 indique que les voxels "haut", "bas", "droite" et "gauche" contiennent moins d'ADN que le voxel "avant". En effet, ces derniers sont des voxels dans lesquels le filament d'ADN change de direction et parcourt donc une distance plus courte de 11 nm réduisant ainsi la quantité d'ADN requise. Les voxels "haut", "bas", "droite" et "gauche" possèdent également un pitch de 17 nm tandis que celui du voxel "avant" est de 10 nm. Cette différence est due au fait qu'avec un pitch de 10 nm, les extrémités des nucléosomes des voxels "haut", "bas", "droite" et "gauche" entrent en collision. Une augmentation du pitch de 7 nm est nécessaire pour éviter cela.

TABLEAU 2.5 – Caractéristiques des 5 voxels implémentés dans DnaFabric : avant, haut, bas, droite et gauche

Caractéristique	Voxels				
	Avant	Haut	Bas	Droite	Gauche
Nombre de nucléosomes	23	11	11	11	11
Nombre de paires de nucléotides	4860	2466	2470	2459	2461
Pitch	10 nm	17 nm	17 nm	17 nm	17 nm
Rayon de l'hélice	10,46 nm	10,46 nm	10,46 nm	10,46 nm	10,46 nm
Nombre de nucléosomes par tour	6	6	6	6	6
Angle delta	60 deg	60 deg	60 deg	60 deg	60 deg
Rayon de la fibre	17 nm	17 nm	17 nm	17 nm	17 nm
Distance parcourue par la fibre	50 nm	39 nm	39 nm	39 nm	39 nm
Longueur des côtés du voxel	50 nm	50 nm	50 nm	50 nm	50 nm
Volume occupé par l'ADN (nm^3)	$5,7 \cdot 10^3$	$2,9 \cdot 10^3$	$2,9 \cdot 10^3$	$2,9 \cdot 10^3$	$2,9 \cdot 10^3$

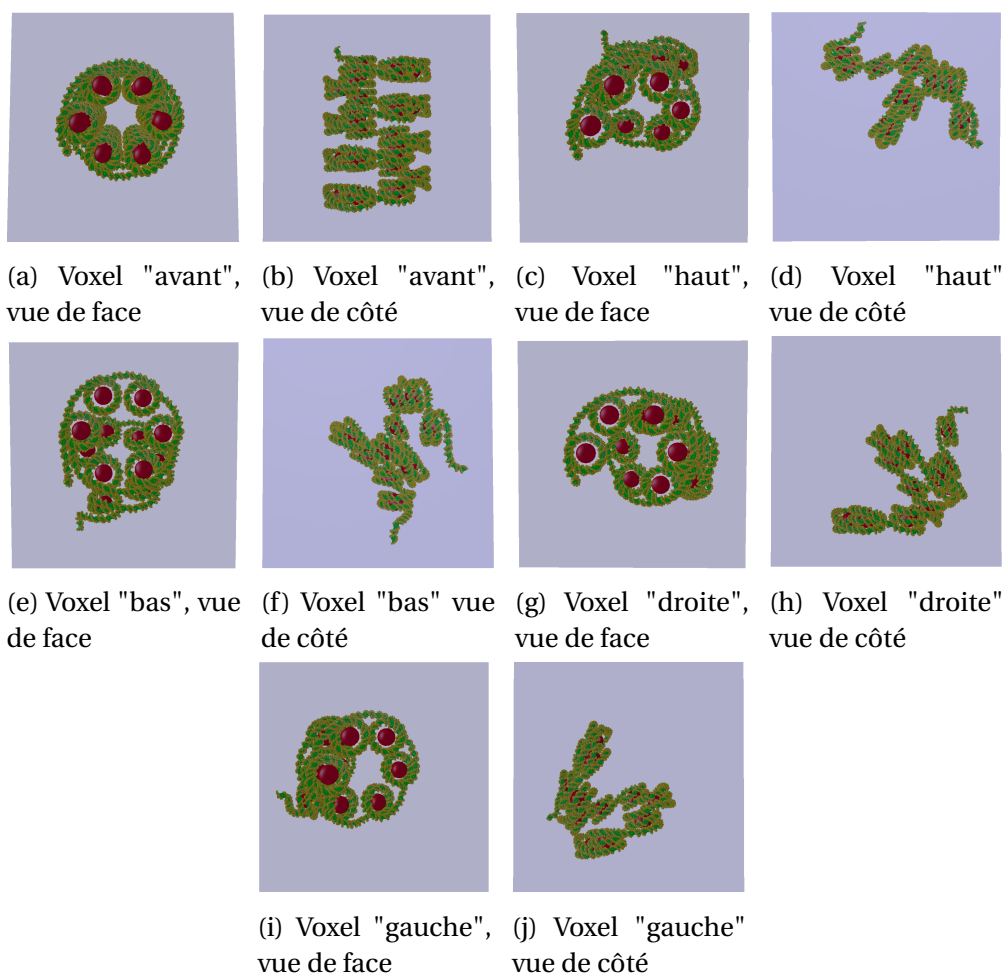


FIGURE 2.18 – Les 5 voxels implémentés dans DnaFabric : avant, haut, bas, droite et gauche.

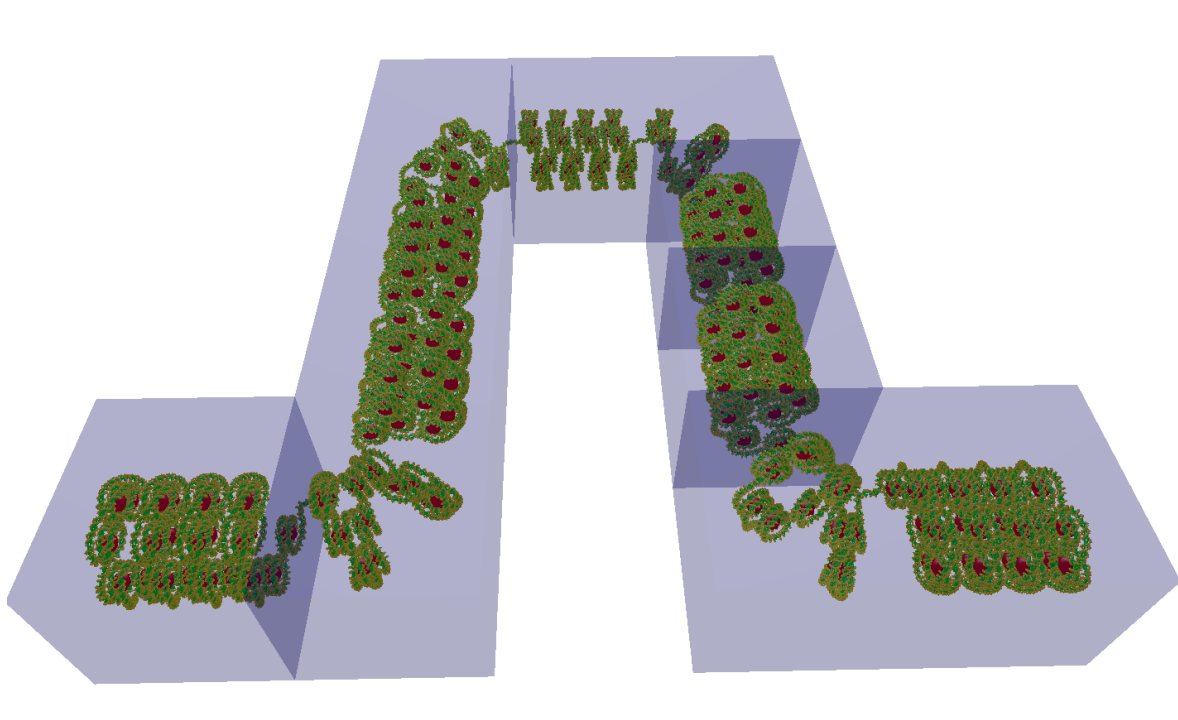


FIGURE 2.19 – Un filament d'ADN constitué de 11 voxels, 205 nucléosomes et 19560 paires de nucléotides.

2.3 Modélisation de noyaux cellulaires

Les modélisations de noyaux cellulaires développées dans ce travail ont pour but la construction de géométries représentant l'ADN du génome humain dans différents noyaux cellulaires. Comme l'agencement de la chromatine dans un noyau varie fortement en fonction des différentes étapes du cycle cellulaire et que les modèles géométriques de noyaux doivent être statiques pour être utilisés avec Geant4, ils ne peuvent être valides que pour une étape donnée. L'étape sur laquelle ce travail se focalise est la phase de quiescence (G0) où la cellule est en état de repos, c'est à dire de non-prolifération. La chromatine est dans un état dit "relaxé" lors de la phase de quiescence. Dans ce travail, les modèles géométriques de noyaux cellulaires (G0, relaxé) ne sont pas directement décrits mais obtenus au moyen d'une simulation dite "de relaxation" [KRETH et al., 2004]. Ainsi, les noyaux sont initialement décrits avec un génome condensé, ce qui correspond à la métaphase, avant d'être relaxés pour obtenir des modèles géométriques représentatifs de la phase de quiescence. Les modélisations effectuées ne prennent en compte que le génome que contiennent les noyaux cellulaires ainsi que toutes les autres entités présentes dans le noyau. Les séquences codantes de l'ADN sont également négligées et remplacées par une simple alternance adénine-thymine puis guanine-cytosine. Précisons cependant que ne prendre en considération que l'ADN dans le noyau cellulaire revient tout de même à inclure quelques $6.4 \cdot 10^9$ paires de nucléotides dans la géométrie. Le projet DnaFabric est précisément conçu dans le but de faciliter la gestion d'un tel nombre d'éléments distincts et organisés hiérarchiquement.

2.3.1 Modélisation du génome humain

Organisation hiérarchique du modèle

Les géométries représentatives du génome humain sont réalisées à l'aide d'une organisation hiérarchique qui étend celle précédemment présentée en 2.2. L'organisation hiérarchique du génome est présentée en figure 2.20. Les éléments de base de cette organisation sont les cinq types de voxels définis en 2.2. Les voxels sont des objets géométriques destinés à être disposés dans d'autres objets géométriques : les domaines. Un domaine est représenté par un objet de forme sphérique et de rayon égal à 500 nm qui contient environ 1.10^6 paires de nucléotides sous la forme de voxels. Les objets "domaines" sont vides lorsqu'ils sont positionnés pour la première fois dans un objet parent et l'ajout de voxels est effectué dans un second temps lors d'un processus de remplissage (cf. 2.3.2). Les objets susceptibles de contenir des domaines sont les objets "chromosomes". Il existe 46 objets "chromosomes" par noyau pour correspondre aux 46 chromosomes du génome humain décrits dans le tableau 2.6 sous la forme de 23 paires de chromosomes. Enfin, les 46 chromosomes sont positionnés à l'intérieur de l'objet géométrique représentant le noyau cellulaire formant ainsi le modèle géométrique d'un noyau. Éventuellement, ce noyau peut ensuite être inclus dans un cytoplasme de manière à former une cellule. Dans cette modélisation, un noyau cellulaire dans un état "relaxé" indique que les chromosomes sont constitués de domaines assemblés sous la forme d'amas et répartis dans l'ensemble du noyau. À l'inverse, dans un état condensé correspondant au début de la métaphase, les domaines sont disposés linéairement tous les 14 nm pour former des chromosomes "condensés". Cette disposition fait apparaître chaque chromosome "condensé" comme un cylindre dont la longueur dépend du nombre de ses domaines.

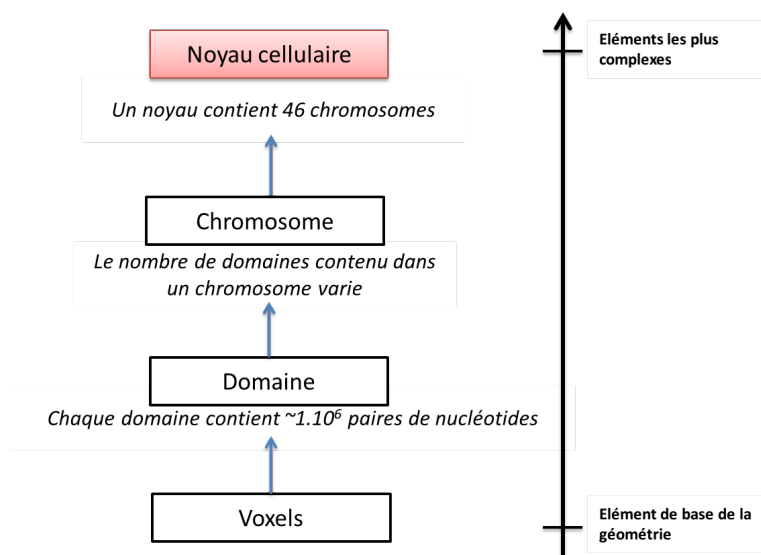


FIGURE 2.20 – L'organisation hiérarchique de l'ADN dans un noyau cellulaire.

Génération d'un génome dans l'état "condensé"

DnaFabric possède un outil dédié à la génération de génomes condensés. Deux informations doivent être fournies à cet outil :

- Le type du génome qui peut être mâle ou femelle.
- La forme et taille du noyau cellulaire dans lequel le génome doit être construit.

TABLEAU 2.6 – Les 23 paires de chromosomes du génome humains.

Paire de chromosomes	Chromosomes	Nombre de domaine
1	1-2	2×250
2	3-4	2×242
3	5-6	2×198
4	7-8	2×190
5	9-10	2×182
6	11-12	2×171
7	13-14	2×159
8	15-16	2×145
9	17-18	2×138
10	19-20	2×134
11	21-22	2×135
12	23-24	2×133
13	25-26	2×114
14	27-28	2×107
15	29-30	2×102
16	31-32	2×90
17	33-34	2×83
18	35-36	2×80
19	37-38	2×59
20	39-40	2×64
21	41-42	2×47
22	43-44	2×51
23	X	156
23	Y	57

La forme du noyau est un élément nécessaire car l'outil de génération positionne aléatoirement les chromosomes et leurs domaines au cours d'un processus itératif où essais et erreurs se succèdent. D'après la littérature, le positionnement des chromosomes dépend de leur taille et les plus longs ont tendance à se trouver vers le centre du noyau [KRETH et al., 2004]. Le processus de génération peut être résumé ainsi :

1. Sélectionner un chromosome parmi les 46.
2. Sélectionner une position aléatoire dans le noyau puis y placer un objet chromosome.
3. Sélectionner une direction aléatoire puis alignement du chromosome selon cette dernière.
4. Placer les domaines de ce chromosome le long de la direction choisie en les espaçant de 15 nm deux à deux et en veillant à ce que le centre du chromosome corresponde au centre du solide formé par l'union de ces domaines. Les domaines peuvent se chevaucher mutuellement et c'est notamment le cas lors de cette configuration.
5. Vérifier que tous les domaines sont dans le noyau. Si ce n'est pas le cas, effacer le chromosome ainsi que ses domaines et retourner à l'étape 2.
6. Vérifier qu'aucun domaine n'entre en collision avec un domaine d'un autre chromosome déjà présent dans le noyau. Si ce n'est pas le cas, effacer le chromosome et ses domaines puis retourner à l'étape 2.

Les étapes 3 à 6 sont rendues possibles par l'association d'une primitive géométrique dite "primitive zéro" à chaque noyau. Cette primitive zéro n'est pas utilisée pour l'affichage 3D mais uniquement pour effectuer des opérations en rapport avec la forme 3D de l'objet concerné. Par exemple, la génération d'une position aléatoire dans un noyau sera prise en charge par la primitive zéro de ce même noyau. En effet, la primitive zéro de ce noyau contient la description la plus détaillée possible de son enveloppe extérieure (si possible une description purement mathématique) et cette description est utilisée pour générer automatiquement des positions aléatoires situées en son sein. En pratique, toute primitive géométrique peut contenir des méthodes (ou fonctions) du type *GenerateRandomPosition()*, *IsInside()* et toute primitive utilisée en tant que "primitive zéro" doit impérativement les avoir. Au final, un exemple de génome condensé généré par DnaFabric est présenté en figure 2.21. En tant qu'objet géométrique, ce génome peut-être sauvegardé dans un fichier *.fab*.

Relaxation de la chromatine

En G0, la chromatine contenue dans le noyau est décondensée et les chromosomes ne sont plus assimilables à des cylindres mais à des amas occupant des territoires chromosomiques dédiés [CREMER et CREMER, 2001; DOS SANTOS, 2013]. La distribution exacte de la chromatine ainsi que les processus impliqués dans sa décondensation constituent un sujet de recherche actuel [BANCAUD et al., 2012; BRANCO et POMBO, 2007; HUET et al., 2014; RAO et al., 2014]. Dans le cadre de ces études, l'ADN est souvent considéré comme un polymère [HALVERSON et al., 2011; VETTOREL et al., 2009] et la distribution finale de la chromatine obtenue au moyen d'une **simulation de relaxation** [BASHNAGEL et al., 2004]. Une simulation similaire [KRETH et al., 2004], mais basée sur la prise en compte de potentiels attractifs et répulsifs au sein d'une simulation utilisant la méthode de Monte-Carlo, est implémentée dans DnaFabric. Cette simulation de relaxation débute à partir d'un génome condensé contenu dans un noyau cellulaire. Ensuite, elle s'exécute sur un thread

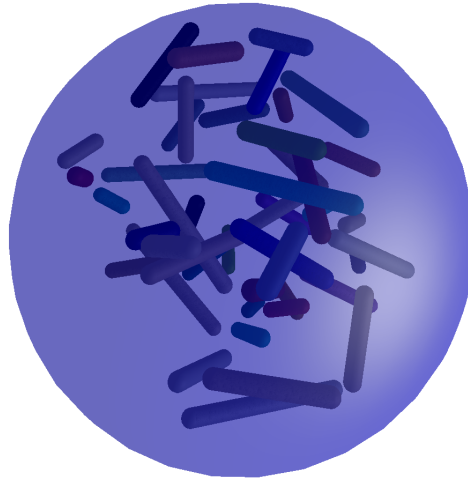


FIGURE 2.21 – Un génome humain de type mâle généré dans un noyau sphérique. Le génome est dans l'état condensé.

dédié (cf. 2.1.3) et les domaines formant les chromosomes se répartissent progressivement dans tout le noyau. Grâce au thread dédié au calcul, l'affichage 3D reste fluide tout au long de la simulation et chaque déplacement de domaine est directement visible. Les simulations de relaxation effectuées dans DnaFabric sont considérées comme terminées après 500000 déplacements de domaine [KRETH et al., 2004]. Un génome similaire à celui du noyau représenté figure 2.21 a été relaxé dans DnaFabric et la figure 2.22 montre le résultat obtenu. Le noyau cellulaire relaxé ne contient, à ce stade, que des chromosomes construits à partir de domaines vides. La nombre de paires de nucléotides du noyau relaxé de la figure 2.22 est donc de zéro.

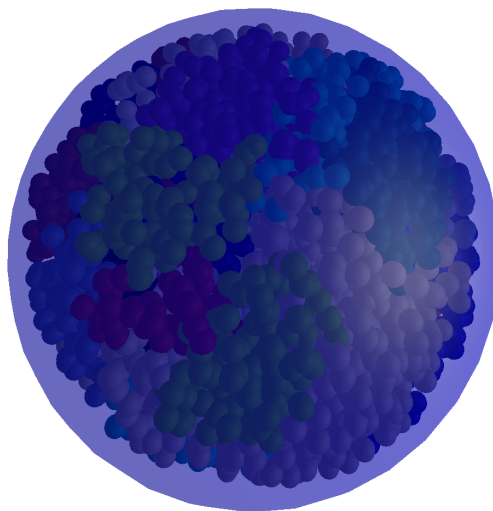


FIGURE 2.22 – Un génome humain de type mâle obtenu après une simulation de relaxation. Le génome est contenu dans un noyau sphérique.

Comme mentionné en section 1.4.2, de récents travaux [BANCAUD et al., 2009, 2012; LIEBERMAN-AIDEN et al., 2009] ont mis en évidence une organisation fractale de la distribution spatiale de la chromatine en G₀, du moins pour certaines échelles spatiales. La publication de ces travaux a éclairé sous un jour nouveau la théorie du "fractal globule", ou "crumpled globule", formulée à la fin des années 80 [GROSBERG et al., 1993, 1988]. Toutefois, la mise en œuvre de ce modèle ne fait pas consensus [BANCAUD et al., 2012; SCHRAM

et al., 2013] et c'est pourquoi il n'a pas été retenu dans ce travail. Ainsi, la distribution de la chromatine illustrée en figure 2.22 n'est pas fractale, tant au niveau du contenu d'un domaine qu'au niveau du noyau dans son ensemble.

2.3.2 Algorithmes de remplissage du noyau

L'ajout de paires de nucléotides dans un noyau cellulaire contenant un génome ne peut avoir lieu dans DnaFabric que si le noyau est relaxé. Un tel noyau est alors pris en charge par un algorithme de remplissage qui y ajoutera des voxels à l'intérieur des domaines (cf. section 2.2.2). Tout comme la simulation de relaxation, l'algorithme de remplissage s'exécute sur un thread dédié de manière à maintenir un affichage 3D fluide tout au long de la simulation. Deux méthodes de remplissage sont implémentées dans DnaFabric qui font toutes deux appel à une même fonction pour générer des filaments d'ADN à partir de voxels : la fonction *GoFromAtoB*(P_A , P_B).

Génération de filaments d'ADN

La génération d'un filament d'ADN peut être faite manuellement, semi-automatiquement ou automatiquement. La génération manuelle consiste à ajouter des objets voxels dans un volume parent tout en veillant à les orienter et positionner correctement. Bien que fonctionnelle, cette méthode d'ajout est fastidieuse et n'empêche pas les chevauchements induits par le positionnement de deux voxels au même endroit. La génération semi-automatique corrige ces défauts avec la création d'un réseau cubique dans lequel les voxels sont positionnés. En effet, chaque case de ce réseau est vérifiée avant chaque placement de voxel et, en cas d'occupation, le placement est annulé. De plus, la génération d'un filament est facilitée par les méthodes *AddVoxelStraight()*, *AddVoxelUp()*, *AddVoxelDown()*, *AddVoxelRight()* et *AddVoxelLeft()* qui prennent en charge l'extension d'un filament à partir d'une de ses extrémités. Enfin, la méthode automatique fait également appel à un réseau cubique dans lequel l'utilisateur peut spécifier une case de départ et une case d'arrivée mais la fonction correspondante est cette fois *GoFromAtoB*(P_A , P_B). DnaFabric générera alors automatiquement un filament d'ADN reliant ces deux cases par le chemin le plus court. L'algorithme 2.3 est chargé de ce processus. Cet algorithme détermine le voxel à ajouter qui, parmi les 5 disponibles, rapprochera le plus l'extrémité du filament de la position d'arrivée. Si ce voxel ne peut être ajouté, à cause d'un obstacle par exemple, alors l'algorithme choisit un voxel au hasard parmi les 5 pour contourner l'obstacle. Le tirage aléatoire peut être répété plusieurs fois d'affilée si l'obstacle est toujours présent.

Deux méthodes distinctes font usage de l'outil de génération de filament d'ADN dans le but de remplir un noyau relaxé avec de l'ADN. La première méthode tente de reproduire des boucles de chromatine dans chaque domaine tandis que la seconde a pour but de maximiser l'occupation de l'espace du noyau par l'ADN.

Méthode de remplissage 1 : les boucles de chromatine

La construction de boucles de chromatine dans chaque domaine du noyau cellulaire tente de reproduire l'organisation de la chromatine observée expérimentalement [RAO et al., 2014]. Cette méthode de remplissage permet la construction d'un filament d'ADN continu pour chaque chromosome. Ce filament passe par chaque domaine du chromosome pour y effectuer des boucles et ceci jusqu'à ce que 1.10^6 paires de nucléotides y soient placées. Ensuite, le filament se dirige vers le domaine suivant. Chacune des

```

Données : Positions de départ et d'arrivée définies dans un réseau cubique
Résultat : Génération d'un filament d'ADN reliant la position de départ  $P_{ini}$  à celle
              d'arrivée  $P_{fin}$ 
si la position de départ correspond à une case occupée du réseau cubique alors
    | Error();
    | Quit();
fin
P =  $P_{ini}$  ;
tant que  $P \neq P_{fin}$  faire
    |  $D = P_{fin} - P$  ; // Direction pointant vers la position d'arrivée
    | /* Tester l'ensemble des directions possibles et déterminer celle qui
    | rapprochera le plus le filament d'ADN de sa position d'arrivée */
    | pour tous les voxels (avant, haut, bas, droite et gauche) faire
    | | Sélectionner un voxel parmi ceux disponibles : avant, haut, bas, droite et
    | | gauche ;
    | |  $D_{test}$  = direction correspondant à celle du voxel sélectionné;
    | |  $proj = D \cdot D_{test}$  ;
    | | Enregistrer  $proj$ ;
    | fin
    | // La direction la plus avantageuse est choisie ici
    | Sélectionner la valeur de  $proj$  la plus faible ainsi que la position et le voxel
    | associé;
    | tant que la position associée est occupée faire
    | | // Tenter un mouvement aléatoire
    | | Sélectionner aléatoirement un voxel parmi ceux disponibles;
    | | Mettre à jour la position associée;
    | fin
    | // La position est libre et le voxel peut être placé
    | Ajouter le voxel avec AddVoxel() et avancer la position P en fonction du voxel
    | ajouté;
fin

```

ALGORITHME 2.3 – Relier deux cases d'un réseau cubique avec un filament d'ADN.

boucles effectuée dans un domaine débute et finit au centre de ce dernier tel qu'illustré en figure 2.23. Les caractéristiques ajustables du processus de construction détaillé en figure 2.23 sont les suivantes :

- La distance P_cP équivalent au rayon du cercle C_1 .
- La distance PI_1 égale à PI_2 .
- Le rayon du cercle C_2 égal aux distances P_cO_1 et P_cO_2 .
- Le nombre de voxels à placer aléatoirement en cas de rencontre d'obstacle. Un obstacle peut être un voxel précédemment positionné. En pratique, plus ce paramètre est élevé et plus les boucles sont larges et étalées.
- Autoriser ou interdire le positionnement de voxels à l'extérieur du domaine. Le fait de positionner des objets géométriques voxels dans un objet domaine implique une relation hiérarchique de type parent-enfant entre le domaine et les voxels mais pas forcément que les voxels soient physiquement dans la sphère représentant le domaine.

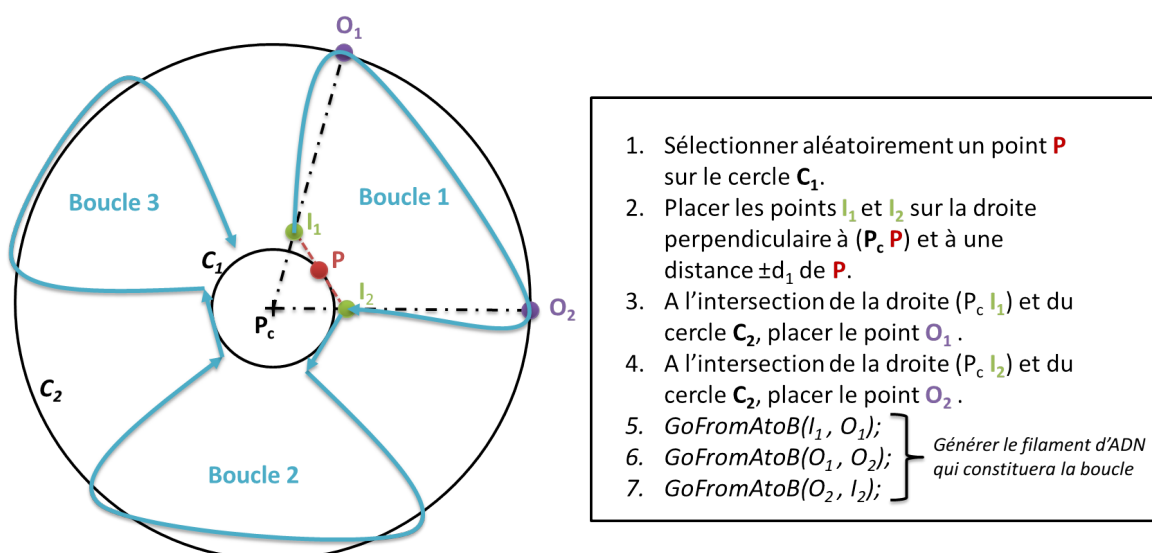


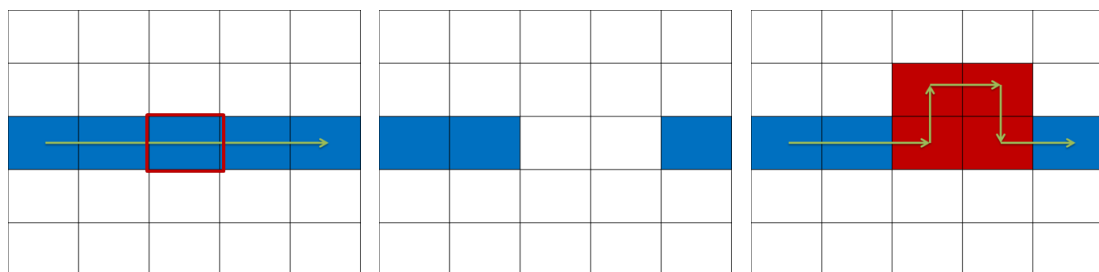
FIGURE 2.23 – Construction d'une boucle de chromatine dans un domaine sphérique représenté par le cercle C_2 . Les valeurs des paramètres employés (C_1 , d_1 ...) dépendent du noyau simulé.

Lorsque le processus de remplissage débute, un algorithme balaie tous les domaines et chromosomes du noyau. Lorsque l'algorithme rencontre le premier domaine d'un chromosome, il démarre la phase de remplissage en effectuant autant de boucles que nécessaire pour atteindre le nombre de 1.10^6 paires de nucléotides dans ce domaine. Toutes les boucles sont constituées à partir du même filament d'ADN fait de voxels attribués au domaine en cours de remplissage. Ensuite, l'algorithme étend le filament en direction du domaine suivant où le processus recommence. Et ce jusqu'à atteindre le dernier domaine du chromosome où l'algorithme arrêtera d'étendre le filament d'ADN. Un nouveau filament d'ADN est créé lors du passage à un nouveau chromosome.

Méthode de remplissage 2 : maximiser la densité d'ADN dans le noyau

Une autre méthode de remplissage consiste à tenter de maximiser la densité d'ADN dans le noyau cellulaire sans se préoccuper de la construction de boucles de chromatine. Bien que ne reposant pas sur une modélisation biologique, cette méthode est implémentée dans DnaFabric en raison de son efficacité lors du remplissage de petits noyaux

compacts tel que le noyau de la cellule endothéliale. En effet, la méthode précédemment décrite requiert un certain espace pour fonctionner et pouvoir positionner aléatoirement les points de départ et d'arrivée des boucles présentes dans chaque domaine. De plus, la présence de nombreux obstacles entre ces différents points, tels que des voxels déjà positionnés par exemple, implique de faire de nombreux écarts aléatoires pour les éviter. Ces écarts allongent très sensiblement la durée de l'algorithme dans de petits noyaux. L'algorithme développé pour maximiser la densité d'ADN minimise ces problèmes au prix de l'abandon de la notion de boucle de chromatine. En effet, cet algorithme commence par relier au moyen d'un filament d'ADN tous les centres des domaines situés dans un même chromosome. Cette opération est rapide car le noyau ne contient pas, à ce moment, d'autres voxels susceptibles de bloquer le filament. Ensuite, l'algorithme va sélectionner aléatoirement un voxel, parmi tous ceux du filament créé, pour le supprimer. Le voxel suivant sera également supprimé de manière à créer un trou de deux "cases" dans le filament d'ADN. Ensuite, le trou est comblé par l'ajout d'un "créneau" constitué de 4 voxels. Cet ajout de 4 voxels pour remplacer les 2 précédemment détruits revient à étendre localement le filament. L'extension ainsi effectuée peut survenir à n'importe quel endroit du filament et pas forcément à l'une de ses extrémités. Ce processus est schématisé en figure 2.24. Lorsque ces extensions se cumulent sur un filament d'ADN donnée alors ce dernier s'étale avec un taux d'occupation de l'espace beaucoup plus important que dans le cas des boucles de chromatine. Le taux d'occupation de l'espace n'est cependant pas équivalent à ceux, pouvant atteindre 100%, des courbes de remplissage de l'espace telle que celle de Hilbert[ROSE, 2001] par exemple. La courbe de Hilbert présente néanmoins le désavantage de ne pouvoir s'adapter à la présence d'obstacles alors que l'algorithme de maximisation développé dans DnaFabric ne rencontre aucune difficulté dans un tel cas.



(a) Sélection aléatoire du 3ème voxel en partant de la droite parmi les voxels du filament d'ADN (cases bleues).
 (b) Suppression du voxel sélectionné ainsi que du voxel suivant.
 (c) Ajout de 4 voxels en remplacement des deux voxels supprimés. Les 4 voxels ajoutés forment un "créneau" (en rouge).

FIGURE 2.24 – Algorithme de remplissage pour maximiser la densité d'ADN.

2.3.3 Les noyaux cellulaires

Visualisation d'un noyau complet

Un noyau relaxé contenant environ $6,4 \cdot 10^9$ paires de nucléotides est visualisable en temps réel à condition de ne pas afficher l'ensemble de ses composants simultanément. L'usage des LODs est particulièrement adapté à une telle situation. Concrètement, il existe 4 primitives géométriques par voxel ce qui équivaut à 4 niveaux de détail par voxel. Les trois premières primitives sont composites et sont construites lors de l'initialisation du

moteur de rendu. Le niveau 1 est le plus détaillé et toutes les paires de nucléotides sont présentes. Le niveau 2 est similaire mais les sphères représentant les molécules sont plus rudimentaires. Le niveau 3 n'affiche qu'une paire de nucléotides sur quatre et ce sous la forme de simples cubes. Enfin, le niveau 4 n'affiche que l'enveloppe externe du voxel et aucune paire de nucléotides. Au delà des voxels, les domaines possèdent deux LODs. Le premier est totalement transparent car il s'agit de laisser s'afficher les différentes représentations 3D des voxels contenus dans chaque domaine. Le second désactive toute représentation des voxels pour n'afficher que les domaines sous la forme de sphères de couleur. Chaque couleur correspondant à un chromosome différent. Enfin, le noyau cellulaire lui-même possède deux LODs : le premier pour afficher ceux de ses composants et le second pour n'afficher que son enveloppe externe. Au final, lorsque qu'une géométrie contenant un noyau cellulaire empli d'ADN est affichée alors l'utilisateur voit apparaître uniquement son enveloppe externe dans un premier temps. Puis, si l'utilisateur approche la vue du noyau, les domaines sphériques apparaîtront avec leurs couleurs permettant de distinguer les différents chromosomes. L'utilisateur peut alors continuer de s'approcher en se dirigeant vers un ensemble de domaines ce qui déclenchera l'apparition des voxels avec différents niveaux de détails. Le processus inverse a lieu si l'utilisateur éloigne la vue du noyau. Le figure 2.25 illustre la représentation 3D d'un noyau cellulaire dans DnaFabric.

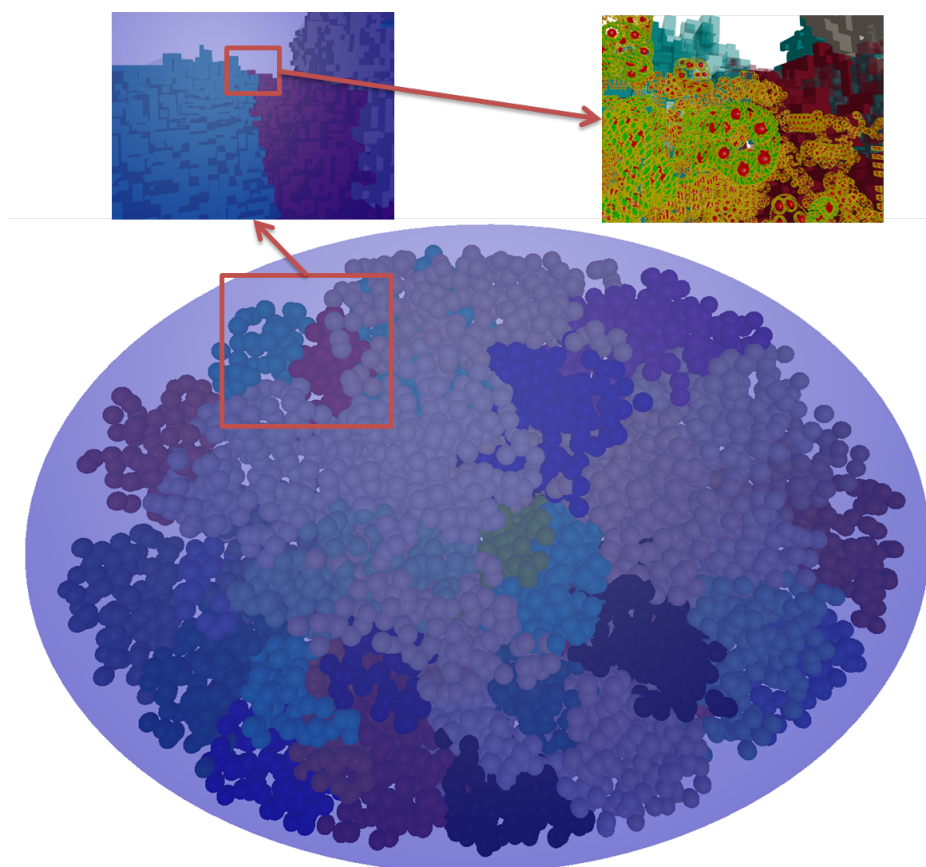


FIGURE 2.25 – Représentation 3D d'un noyau cellulaire au travers de trois niveaux de zoom.

Les différents noyaux générés avec DnaFabric

Trois modèles de noyaux cellulaires ont été générés dans le cadre de ce travail de thèse et leurs principales caractéristiques sont présentées dans le tableau 2.7. Les types

de noyaux créés sont les suivants :

- **Fibroblaste** représenté en figure 2.26.
- **Lymphocyte** représenté en figure 2.27.
- **Endothélium** représenté en figure 2.28.

Chacun de ces types se décline en quatre versions correspondant aux différents états de son génome (condensé ou relaxé) ainsi qu'à son genre (mâle ou femelle). Les 12 versions ainsi produites sont enregistrées dans des fichiers ".fab" en tant que modèles géométriques accessibles avec DnaFabric et dans des fichiers ".fab2g4dna" en tant que géométries dédiées au calcul avec Geant4.

TABLEAU 2.7 – Caractéristiques des noyaux générés avec DnaFabric.

Cellule	Forme	Dimensions
Fibroblaste	Ellipsoïde	semi-major : $9,85 \cdot 10^3$ nm semi-minor ₁ : $7,1 \cdot 10^3$ nm semi-minor ₂ : $2,5 \cdot 10^3$ nm
Lymphocyte	Sphère	rayon : $5,0 \cdot 10^3$ nm
Endothélium	Cylindre elliptique	semi-major : $9,5 \cdot 10^3$ nm semi-minor : $5,1 \cdot 10^3$ nm hauteur : $1,0 \cdot 10^3$ nm

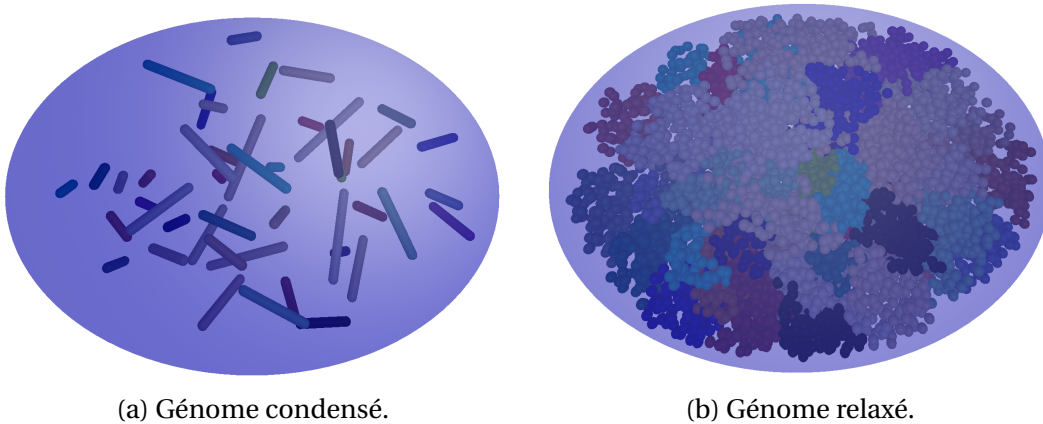


FIGURE 2.26 – Modélisation d'un noyau de cellule fibroblaste de forme elliptique.

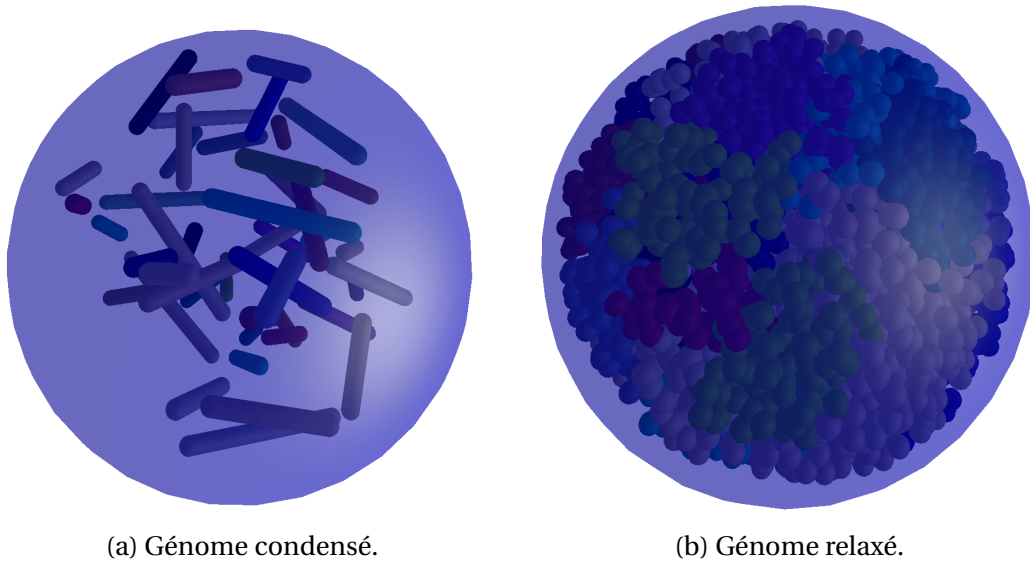


FIGURE 2.27 – Modélisation d'un noyau de cellule lymphocyte de forme sphérique.

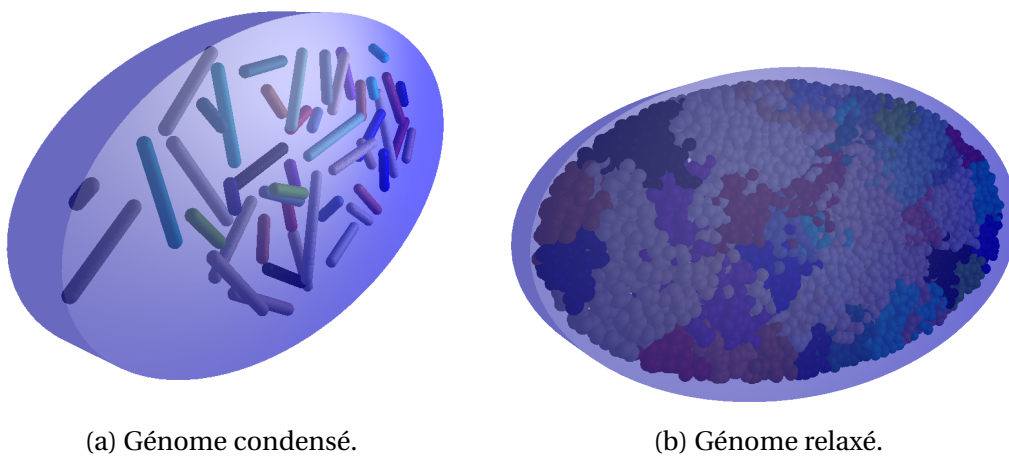


FIGURE 2.28 – Modélisation d'un noyau de cellule endothéliale en forme de cylindre elliptique.

Références

- ALIZADEH, E., A. G. SANZ, G. GARCIA et L. SANCHE. 2013, «Radiation Damage to DNA : The Indirect Effect of Low Energy Electrons», *J. Phys. Chem Lett*, vol. 4, n° 5, p. 820–825. [74](#)
- BANCAUD, A., S. HUET, N. DAIGLE, J. MOZZICONACCI, J. BEAUDOUIN et J. ELLENBERG. 2009, «Molecular crowding affects diffusion and binding of nuclear proteins in heterochromatin and reveals the fractal organization of chromatin», *The EMBO journal*, vol. 28, n° 24, p. 3785–3798. [85](#)
- BANCAUD, A., C. LAVELLE, S. HUET et J. ELLENBERG. 2012, «A fractal model for nuclear organization: current evidence and biological implications», *Nucleic Acids Research*, vol. 40, n° 18, p. 8783–8792. [84](#), [85](#)
- BASHNAGEL, J., J. P. WITTMER et H. MEYER. 2004, *Monte Carlo Simulation of Polymers: Coarse-Grained Models*, n° 23 dans NIC series, NIC, Jülich, ISBN 978-3-00-012641-3. [84](#)
- BERNAL, M., D. SIKANSI, F. CAVALCANTE, S. INCERTI, C. CHAMPION, V. IVANCHENKO et Z. FRANCIS. 2013, «An atomistic geometrical model of the B-DNA configuration for DNA–radiation interaction simulations», *Computer Physics Communications*, vol. 184, n° 12, p. 2840–2847. [73](#), [75](#), [76](#)
- BERNAL, M. A., D. SIKANSI, F. CAVALCANTE, S. INCERTI, C. CHAMPION, V. IVANCHENKO, Z. FRANCIS et M. KARAMITROS. 2014, «Performance of a new atomistic geometrical model of the B-DNA configuration for DNA-radiation interaction simulations», *Journal of Physics : Conference Series*, vol. 490, p. 012 150. [57](#)
- BILJECKI, F., H. LEDOUX, J. STOTER et J. ZHAO. 2014, «Formalisation of the level of detail in 3d city modelling», *Computers, Environment and Urban Systems*, vol. 48, p. 1–15. [68](#)
- BRANCO, M. R. et A. POMBO. 2007, «Chromosome organization: new facts, new models», *Trends in Cell Biology*, vol. 17, n° 3, p. 127–134. [57](#), [78](#), [84](#)
- BUG, M. U., E. SURDUTOVICH, H. RABUS, A. B. ROSENFELD et A. V. SOLOV'YOV. 2012, «Nanoscale characterization of ion tracks : MC simulations versus analytical approach», [57](#)
- CREMER, T. et C. CREMER. 2001, «Chromosome territories, nuclear architecture and gene regulation in mammalian cells», *Nature reviews genetics*, vol. 2, n° 4, p. 292–301. [84](#)
- DILLON, N. 2004, «Heterochromatin structure and function», *Biology of the Cell*, vol. 96, n° 8, p. 631–637. [78](#)
- DOS SANTOS, M. 2013, *Modélisation de la topologie des dépôts d'énergie créés par un rayonnement ionisant à l'échelle nanométrique dans les noyaux cellulaires et relation avec les événements précoces radio-induits*, thèse de doctorat, Bordeaux 1, Ecole doctorale des sciences physiques et de l'ingénieur. [84](#)
- FRIEDLAND, W., M. DINGFELDER, P. KUNDRÁT et P. JACOB. 2011, «Track structures, DNA targets and radiation effects in the biophysical Monte Carlo simulation code PARTRAC», *Mutation Research/Fundamental and Molecular Mechanisms of Mutagenesis*, vol. 711, n° 1-2, p. 28–40. [57](#), [74](#)

- FRIEDLAND, W., H. G. PARETZKE, F. BALLARINI, A. OTTOLENGHI, G. KRETH et C. CREMER. 2008, «First steps towards systems radiation biology studies concerned with DNA and chromosome structure within living cells», *Radiation and Environmental Biophysics*, vol. 47, n° 1, p. 49–61. [78](#)
- GARTY, G., R. SCHULTE, S. SHCHEMELININ, C. LELOUP, G. ASSAF, A. BRESKIN, R. CHECHIK, V. BASHKIROV, J. MILLIGAN et B. GROSSWENDT. 2010, «A nanodosimetric model of radiation-induced clustered DNA damage yields», *Physics in medicine and biology*, vol. 55, n° 3, p. 761. [57](#)
- GROBERG, A., Y. RABIN, S. HAVLIN et A. NEER. 1993, «Crumpled globule model of the three-dimensional structure of DNA», *EPL (Europhysics Letters)*, vol. 23, n° 5, p. 373. [85](#)
- GROBERG, A. Y., S. K. NECHAEV et E. I. SHAKHNOVICH. 1988, «The role of topological constraints in the kinetics of collapse of macromolecules», *Journal de physique*, vol. 49, n° 12, p. 2095–2100. [85](#)
- HALVERSON, J. D., W. B. LEE, G. S. GRETT, A. Y. GROBERG et K. KREMER. 2011, «Molecular dynamics simulation study of nonconcatenated ring polymers in a melt. I. Statics», *The Journal of chemical physics*, vol. 134, n° 20, p. 204904. [84](#)
- HORNUS, S., B. LÉVY, D. LARIVIÈRE et E. FOURMENTIN. 2013, «Easy dna modeling and more with graphitelifeexplorer», *PloS one*, vol. 8, n° 1, p. e53609. [57](#)
- HUET, S., C. LAVELLE, H. RANÇON, P. CARRIVAIN, J.-M. VICTOR et A. BANCAUD. 2014, «Relevance and Limitations of Crowding, Fractal, and Polymer Models to Describe Nuclear Architecture», dans *International Review of Cell and Molecular Biology*, vol. 307, Elsevier, ISBN 978-0-12-800046-5, p. 443–479. [84](#)
- KORNBERG, R. D. 1974, «Chromatin structure : a repeating unit of histones and DNA», *Science*, vol. 184, n° 4139, p. 868–871. [76](#)
- KRETH, G., J. FINSTERLE, J. VON HASE, M. CREMER et C. CREMER. 2004, «Radial arrangement of chromosome territories in human cell nuclei : a computer model approach based on gene density indicates a probabilistic global positioning code», *Biophysical journal*, vol. 86, n° 5, p. 2803–2812. [81](#), [84](#), [85](#)
- LESLIE, A., S. ARNOTT, R. CHANDRASEKARAN et R. RATLIFF. 1980, «Polymorphisme of DNA double helices», *J. Mol. Biol.*, vol. 143, p. 49–72. [75](#)
- LIEBERMAN-AIDEN, E., N. L. VAN BERKUM, L. WILLIAMS, M. IMAKAEV, T. RAGOCZY, A. TELLING, I. AMIT, B. R. LAJOIE, P. J. SABO, M. O. DORSCHNER, R. SANDSTROM, B. BERNSTEIN, M. A. BENDER, M. GROUDINE, A. GNIRKE, J. STAMATOYANNOPOULOS, L. A. MIRNY, E. S. LANDER et J. DEKKER. 2009, «Comprehensive Mapping of Long-Range Interactions Reveals Folding Principles of the Human Genome», *Science*, vol. 326, n° 5950, p. 289–293. [85](#)
- MEYLAN, S., U. VIMONT, S. INCERTI, I. CLAIRAND et C. VILLAGRASA. 2016, «Geant4-DNA simulations using complex DNA geometries generated by the DnaFabric tool», *Computer Physics Communications*, vol. 204, p. 159–169. [78](#)
- RAO, S., M. HUNTLEY, N. DURAND, E. STAMENOVA, I. BOCHKOV, J. ROBINSON, A. SANBORN, I. MACHOL, A. OMER, E. LANDER et E. AIDEN. 2014, «A 3d Map of the Human

Genome at Kilobase Resolution Reveals Principles of Chromatin Looping», *Cell*, vol. 159, n° 7, p. 1665–1680. [84](#), [86](#)

ROSE, N. J. 2001, *Hilbert-type space-filling curves*. [89](#)

SCHRAM, R. D., G. T. BARKEMA et H. SCHIESSEL. 2013, «On the stability of fractal globules», *The Journal of chemical physics*, vol. 138, n° 22, p. 224 901. [85](#)

VETTOREL, T., A. Y. GROSBERG et K. KREMER. 2009, «Statistics of polymer rings in the melt : a numerical simulation study», *Physical Biology*, vol. 6, n° 2, p. 025 013. [84](#)

Chapitre 3

Simulation de l'irradiation d'un noyau cellulaire

« On en fait jamais attention à ce qui a été fait ; on ne voit que ce qui reste à faire. »

Marie Curie

Sommaire

3.1 Simulation de l'étape physique	99
3.1.1 Prise en compte de l'ADN lors du transport des particules	99
3.1.2 Choix d'une méthode pour l'attribution d'une interaction à un composant de l'ADN	107
3.1.3 Introduction du noyau cellulaire dans la simulation	110
3.1.4 Génération de données lors de la simulation	113
3.2 Simulation des étapes physico-chimique et chimique	116
3.2.1 Prise en compte du modèle géométrique	116
3.2.2 Génération de données lors de la simulation	120
3.2.3 Quelques paramètres spécifiques à l'étape chimique	121
3.3 Organisation de la simulation des trois étapes	122
3.3.1 Contenu et architecture	122
3.3.2 Utilisation de "FullSim"	124
Références	130

Le développement, dans le cadre de ce travail thèse, d'un outil capable de simuler les étapes physique, physico-chimique et chimique consécutives à l'irradiation d'un noyau cellulaire a consisté à créer différents programmes de simulation réunis au sein d'une **chaîne de calcul** nommée **FullSim**. Le travail de développement s'est déroulé en trois parties principales : création d'une application utilisateur Geant4-DNA pour la simulation de l'étape physique, création d'une application utilisateur Geant4-DNA pour la simulation des étapes physico-chimique et chimique et, enfin, mise en place des routines d'analyse et de synchronisation de l'ensemble des programmes impliqués. Les éléments présents dans FullSim utilisent directement Geant4.LDRI.01, ROOT-CERN et, indirectement, le code DnaFabric. Geant4.LDRI.01 est une version dérivée de Geant4.10.1 spécifiquement développée dans le cadre de ce travail de thèse.

3.1 Simulation de l'étape physique

La simulation de l'étape physique s'effectue au moyen d'une application utilisateur nommée **phys_geo** qui est construite selon l'architecture mentionnée en section 1.2.4. Cette application utilisateur permet, entre autres, de générer une géométrie Geant4 à partir d'un fichier descriptif issu de DnaFabric (.fab2g4dna). **phys_geo** fonctionne avec Geant4.LDRI.01 qui présente des spécificités vis-à-vis de Geant4. Tout d'abord, Geant4.LDRI.01 permet à Geant4-DNA de prendre en compte des matériaux représentatifs de l'ADN lors du transport de particules, ce qui n'est pas faisable par défaut avec une version classique de Geant4-DNA. En effet, Geant4-DNA ne transporte que des particules dans l'eau liquide dans sa configuration de base. Ensuite, Geant4.LDRI.01 offre des capacités de paramétrisation étendues par rapport à la version publique de Geant4. Concrètement, Geant4 permet uniquement de paramétrer des volumes identiques dans un volume parent donné par défaut. Néanmoins, la modification manuelle d'une variable (pointeur sur le volume logique) rend possible l'intégration de volumes distincts au sein d'une même paramétrisation. Cela est particulièrement pratique pour la paramétrisation des cinq voxels introduits dans le chapitre précédent. Cependant, la paramétrisation de volumes distincts avec Geant4 ne fonctionne que dans le cadre d'une simulation séquentielle, c'est à dire non parallélisée, limitant ainsi la rapidité du calcul. C'est à ce niveau que Geant4.LDRI.01 intègre des modifications pour rendre compatible l'exécution d'une simulation en parallèle (mode MT) avec une paramétrisation intégrant des volumes distincts. Cette section, consacrée à la simulation de l'étape physique, détaille la prise en compte des matériaux représentatifs de l'ADN, l'introduction d'un noyau cellulaire dans la simulation et, enfin, les données générées lors de cette simulation.

3.1.1 Prise en compte de l'ADN lors du transport des particules

Nouvelles sections efficaces pour les constituants de l'ADN

Comme indiqué en section 1.2.3, la plupart des codes de Monte-Carlo de structure de trace qui mettent en œuvre une approche "pas à pas" (discrète) en radiobiologie considèrent que le milieu cellulaire est assimilable à de l'eau liquide, ADN compris. Ceci signifie que les sections efficaces utilisées pour le transport des particules sont toujours celles de l'eau liquide. La géométrie représentative de l'ADN est généralement introduite dans un second temps et superposée à une carte en 3D contenant l'ensemble des interactions enregistrées entre les particules transportées et l'eau liquide. Cette superposition permet d'associer à chaque interaction physique un volume d'occurrence après chaque

calcul. La situation est cependant différente avec Geant4 car ce dernier permet de créer et d'introduire facilement des volumes géométriquement complexes dans une simulation [DOS SANTOS, 2013]. Dans ce contexte, il est alors plus aisé de définir des volumes pris directement en compte lors de la simulation physique et de leur associer de l'eau liquide comme matériau. En pratique les deux approches génèrent des résultats identiques mais l'utilisation de Geant4 offre la possibilité d'associer aisément des matériaux différents aux volumes présents dans la simulation.

La prise en compte d'un nouveau matériau dans un code de Monte-Carlo correspond à l'introduction de nouvelles sections efficaces lors du calcul du libre parcours moyen (cf. section 1.2.1). Ces sections efficaces pouvant être obtenues au moyen de modèles physiques ou d'expérimentations. Concernant l'ADN, des mesures et analyses ont été récemment effectuées au Physikalisch-Technische Bundesanstalt (PTB, Allemagne) pour des électrons et des protons avec des énergies allant respectivement de 12 eV à 1 keV et de 70 keV à 10 MeV. Les mesures concernent le tetrahydrofuran (THF) [BAEK et al., 2012] et la pyrimidine (PY) [BAEK et al., 2013]. A partir de ces mesures, un nouvel ensemble de sections efficaces a été calculé pour les ionisations, les interactions élastiques et les excitations [BUG, 2014]. Ces sections efficaces ont ensuite été extrapolées à la purine (PU) et au timéthylphosphate (TMP) puis l'ensemble des données ont été introduites dans le code de Monte Carlo P'Tra [GROSSWENDT, 2002] développé au PTB. Les mesures n'ont pas été effectuées directement sur les constituants de l'ADN (2-désoxyribose, phosphate, adénine, guanine, cytosine et thymine) en raison de contraintes expérimentales [BUG, 2014]. Cependant, la combinaison de THF avec TMP forme un précurseur de la 2-désoxyribose et du phosphate tandis que PU et PY sont des précurseurs des bases de l'ADN (adénine, guanine, thymine et cytosine). Le fait que THF, TMP, PY et PU soient des précurseurs des constituants de l'ADN pourrait en faire des substituts acceptables aux constituants de l'ADN dans le cadre d'une simulation Monte Carlo. Néanmoins, le choix a été fait dans ce travail d'appliquer un facteur à chaque jeu de sections efficaces de manière à obtenir une meilleure approximation des sections efficaces des constituants de l'ADN. Les facteurs utilisés correspondent au ratio du nombre d'électrons de valence d'un constituant de l'ADN sur le nombre d'électrons de valence du précurseur lui correspondant [BUG, 2014; LINERT et al., 2012; SHAFRANYOSH et al., 2006]. Les valeurs des facteurs ainsi utilisés sont présentées dans le tableau 3.1. le tableau 3.2 fournit des informations concernant les couches électroniques des matériaux précédemment mentionnés. Finalement et à l'aide de cette procédure, les sections efficaces de dix matériaux distincts sont obtenues : THF, TMP, PU, PY, adénine, guanine, cytosine, thymine, 2-désoxyribose et phosphate. Ces sections efficaces sont illustrées en annexe A pour les électrons et en annexe B pour les protons.

Développement d'un nouveau système de gestion des matériaux dans Geant4-DNA

Les sections efficaces mentionnées précédemment ont été introduites dans Geant4.-LDRI.01 sous la forme de tables de données lors de ce travail de thèse. Ainsi, le transport d'électrons de 12 eV et 1 keV et de protons de 70 à 10 MeV peut être effectué dans les matériaux THF, TMP, PU, PY, adénine, guanine, cytosine, thymine, 2-désoxyribose et phosphate. Les interactions physiques caractérisées par ces sections efficaces sont les ionisations (électrons et protons) ainsi que les collisions élastiques et excitations (électrons). Notons que les sections efficaces de l'hydrogène, des changements de charge et des excitations concernant les protons ne sont pas introduites dans Geant4 et que les interactions associées seront négligées dans les simulations impliquant le transport de protons. Une modélisation simplifiée concernant l'effet Auger a également été incluse. Dans cette

TABLEAU 3.1 – Les facteurs appliqués aux sections efficaces des précurseurs pour obtenir celles des constituants de l'ADN. Les facteurs sont calculés en effectuant le rapport entre le nombre d'électrons de valence d'un constituant de l'ADN et celui du précurseur lui correspondant.

DNA	Précurseur	Facteur
2-Désoxyribose	THF	33/30
Acide phosphorique	TMP	33/50
Cytosine	PY	42/30
Thymine	PY	48/30
Adénine	PU	50/44
Guanine	PU	56/44

TABLEAU 3.2 – Informations concernant les couches électroniques des constituants de l'ADN et de leurs précurseurs.

Matériau	Nb couches	Énergie de liaison (eV)	
		Couche externe	Couche profonde
Eau	5	10.79	539.0
THF	20	9.74	557.94
TMP	37	10.81	2178.05
PY	21	9.73	423.44
PY	21	9.73	423.44
PU	31	9.58	423.98
PU	31	9.58	423.98

dernière, toute ionisation de la couche K d'un atome de carbone ou d'oxygène entraîne l'émission d'un électron Auger tandis que l'ionisation de la couche K d'un atome de nitrogène entraîne celle de deux électrons. Finalement, quatre nouveaux modèles ont été introduits dans Geant4-DNA (Geant4.LDRI.01) pour utiliser les sections efficaces introduites sous la forme de tables :

- G4DNAPTBElasticModel
- G4DNAPTBIonisationModel
- G4DNAPTBExcitationModel
- G4DNAPTBAugerModel

Cependant, historiquement, l'extension Geant4-DNA a été conçue pour effectuer le transport de particules uniquement dans de l'eau liquide. Cet état de fait ne se traduit pas uniquement par l'absence de données et de modèles concernant d'autres matériaux mais également par l'absence d'un système unifié permettant la gestion de différents matériaux dans une simulation. Ce dernier point s'est avéré particulièrement problématique dans le cadre de ce travail et un système de gestion des matériaux dédié a été développé.

Lors du transport des particules, le fonctionnement classique de Geant4 et Geant4-DNA est celui représenté en section 1.2.4, figure 1.10. Lors du calcul du step, une interaction est choisie via une sélection basée sur les libres parcours moyens retournés par les différents processus de Geant4-DNA (ionisation, élastique, excitation ...). Pour cela, les processus utilisent les modèles physiques inclus dans Geant4-DNA. Les libres parcours moyens sont obtenus (cf. section 1.2.2) à partir de la section efficace retournée par la fonction *CrossSectionPerVolume()* qui doit être présente dans toute implémentation d'un modèle physique. Une fois le type de processus choisi, l'état de la particule transportée est modifié par le processus retenu au moyen d'un appel à la fonction *SampleSecondaries()* également présente dans toute implémentation d'un modèle physique. Cette fonction prend en charge les pertes d'énergie et la création d'éventuelles particules secondaires. La fonction *CrossSectionPerVolume()* retourne 0 si le matériau dans lequel la particule est transporté n'est pas de l'eau liquide. Une section efficace valant 0 génère un libre parcours moyen très élevé ce qui rend la sélection du processus concerné très peu probable. En utilisant cela, une façon simple d'introduire les matériaux de type ADN dans Geant4-DNA serait d'attribuer à chaque nouveau modèle PTB une méthode *CrossSectionPerVolume()* retournant une section efficace nulle si le matériau dans lequel la particule est transportée ne correspond pas à celui pris en charge par le modèle. Bien que fonctionnelle, cette approche présente cependant un défaut : elle ne permet pas de prendre en compte des matériaux composites. En effet, les sections efficaces utilisées pour prendre en compte l'ADN dans notre simulation sont une combinaison des sections efficaces des différents composants de l'ADN (cf. tableau 3.3). Ainsi, par exemple, dans le cas d'un matériau composite constitué à 50% de THF et à 50% d'eau liquide, il est nécessaire de combiner les sections efficaces des deux composants pour calculer une section efficace que nous qualifierons de *combinée*. Les sections efficaces des composants sont alors calculées par des modèles différents, G4DNABornIonisationModel dans le cas de l'eau et G4DNAPTBIonisationModel pour le THF, rendant impossible leur combinaison directe au sein d'un même modèle. A ce niveau, une façon de contourner le fait que les modèles soient différents serait de multiplier la section efficace retournée par chaque modèle par la densité du composant associé et d'activer les deux modèles simultanément. Une telle activation simultanée requiert l'attribution d'un processus Geant4 à chaque modèle, et donc à chaque composant du matériau composite, ce qui a pour effet de rompre l'équivalence entre processus et interaction physique propre à Geant4 car il existerait alors plu-

sieurs processus pour un seul type d'interaction physique. Finalement, pour conserver l'équivalence entre processus et interaction physique et prendre en compte des matériaux composite, la combinaison des sections efficaces doit être effectuée par un élément tierce qui ne soit pas un modèle.

La solution finalement retenue dans ce travail est d'introduire une classe intermédiaire, appelée *G4DNAModelInterface*, pour faire office de tampon entre le processus et les modèles physiques. Ainsi, plusieurs modèles prenant en charge les ionisations dans différents matériaux peuvent être associés à une *G4DNAModelInterface* attribuée au processus d'ionisation. De cette manière, l'équivalence entre interaction physique et processus est maintenue. Avec cette solution, chaque processus de Geant4-DNA possède une instance de *G4DNAModelInterface* qui se charge d'analyser le matériau dans lequel se trouve la particule transportée, d'identifier ses éventuels sous-composants, de sélectionner les modèles physiques associés et, enfin, d'appeler les fonctions *CrossSectionPerVolume()* et *SampleSecondaries()* **des modèles**. Suite aux appels de ces fonctions *CrossSectionPerVolume()*, la section efficace combinée est calculée puis transmise au processus. Le processus retourne ensuite un libre parcours moyen directement issu de cette section efficace permettant ainsi à la sélection aléatoire d'un processus parmi ceux disponibles d'avoir lieu. Plus formellement et dans le cas d'un matériau composite, les étapes associées à ce processus sont les suivantes :

1. Les constituants du matériau composite sont définis individuellement dans l'application utilisateur et une fraction massique est donnée pour chaque constituant (ie. composant). La masse volumique du matériau composite est également fournie ainsi que la composition atomique de chaque composant.
2. Le code calcule le nombre de molécules par unité de volume pour chacun des composants à l'aide de l'équation 3.1.

$$n_{composant} = \frac{\rho_{composite}}{M_{composite}} \times N_A \times F_{composant} \quad (3.1)$$

avec $n_{composant}$ le nombre de molécules par unité de volume d'un composant donné, $\rho_{composite}$ la densité du matériau composite, $M_{composite}$ la masse molaire du matériau composite calculée à partir de sa composition atomique et de sa densité, N_A le nombre d'Avogadro et $F_{composant}$ la fraction massique du composant.

3. Lors d'une simulation Geant4, un tirage aléatoire est effectué à chaque step pour sélectionner une longueur à parcourir et un type d'interaction (processus Geant4). Dans le cas d'un matériau composite, le libre parcours moyen est calculé avec l'équation 3.2 pour chacune des interactions physiques.

$$\lambda_{inter} = \frac{1}{\sum_{k=1}^N (\sigma_k \times n_{composant})} \quad (3.2)$$

Avec λ_{inter} le libre parcours moyen associé à une interaction physique, N le nombre de composants dans le matériau composite, σ la section efficace qui dépend des caractéristiques de la particule transportée (énergie cinétique ...) et du matériau k dans lequel elle évolue. Une fois les λ_{inter} calculés, le code sélectionne un processus et une longueur à parcourir en effectuant un tirage aléatoire sur ces derniers.

4. Après avoir sélectionné un processus et une longueur à parcourir lors du step, les éventuelles particules secondaires produites ainsi que les pertes d'énergie de la particule transportée doivent être prises en compte. Ces éléments dépendent des caractéristiques de la particule transportée et aussi des caractéristiques du milieu.

Le milieu étant composite dans notre cas, ces informations ne sont disponibles que pour chacun des composants. Un composant doit donc être choisi et cela est effectué au moyen d'un tirage aléatoire basé sur des probabilités calculées avec l'équation 3.3.

$$p_k = \sigma_k \times n_{composant} \quad (3.3)$$

avec p_k la probabilité de choisir le composant k .

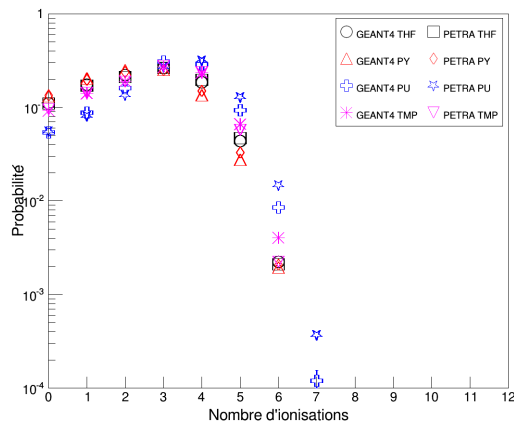
Vérification du système

L'implémentation du système de gestion des matériaux précédemment décrit (cf. section 3.1.1) a été vérifiée par une succession de simulations effectuées conjointement avec Geant4.LDRI.01 et PTra en utilisant des matériaux simples (non composites). La vérification des modèles physiques a été effectuée au moyen d'une série de simulations lancées avec PTra dans un premier temps et Geant4 ensuite. Les caractéristiques de ces simulations étaient :

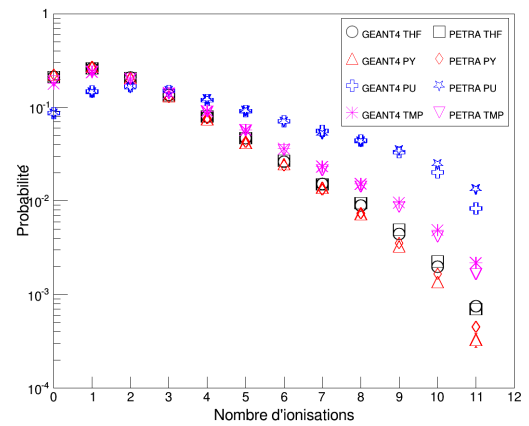
- Géométrie : un volume "world" cubique de 10 nm de côté contenant un cylindre de 2,3 nm de diamètre et 3,4 nm de hauteur.
- Source : les particules primaires sont des électrons mono-énergétiques tirés en direction du centre du cylindre depuis un point situé à sa mi-hauteur et sur sa surface. Les énergies utilisées vont de 50 eV à 1000 eV.
- Matériau : le cylindre précédemment décrit est successivement rempli par les matériaux précurseurs de l'ADN : THF, TMP, PY et PU. A chaque remplissage correspond une série de simulations lancées avec des électrons primaires de différentes énergies.
- Résultat : Le nombre d'ionisations générées dans le cylindre est enregistré par particule primaire dans le but de générer une distribution de probabilité. Cette distribution de probabilité représente la probabilité d'obtenir un nombre donné d'ionisations suite au lancement d'une particule primaire d'énergie E dans un cylindre rempli d'un matériau m .

La figure 3.1 présente les distributions de probabilité calculées avec Geant4 et PTra pour des électrons primaires de 100, 250, 500 et 1000 eV. Les probabilités calculées avec Geant4 et PTra sont identiques pour des nombres d'ionisations inférieurs à 4 ionisations pour 100 eV, à 10 pour 250 eV et 500 eV et à 7 pour 1000 eV. Concernant les tailles de cluster supérieures à ces limites, les probabilités sont sous-estimées par Geant4 pour 100 et 250 eV et, à l'inverse, surestimées pour 500 et 1000 eV. Toutefois, les écarts sont toujours inférieurs à 0,1 dans l'absolu ce qui est considéré comme acceptable, indiquant ainsi que les modèles physiques implémentés dans Geant4 produisent des résultats similaires à ceux du code PTra. L'implémentation des modèles physiques pour les précurseurs de l'ADN est donc considérée comme vérifiée. Cette vérification s'étend également aux constituants de l'ADN pris séparément car ils sont pris en charge par les mêmes modèles que les précurseurs de l'ADN.

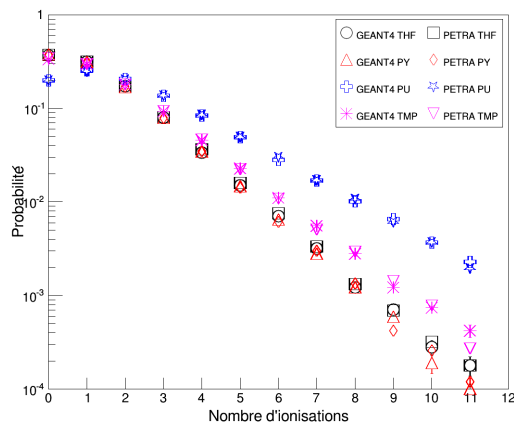
L'implémentation du système de management des matériaux a été testée de façon similaire à celle des modèles physiques, à ceci près que les dimensions du cylindre utilisé sont de 1,8 nm de diamètre et 4,3 nm de hauteur. Ce cylindre est également rempli d'un unique matériau composite décrit dans le tableau 3.3. Ce matériau composite est construit de manière à être représentatif de l'ADN et de la couche d'hydratation qui l'entoure (cf. section 2.2.1). Sa densité est calculée à partir de l'équation 3.4 et est de



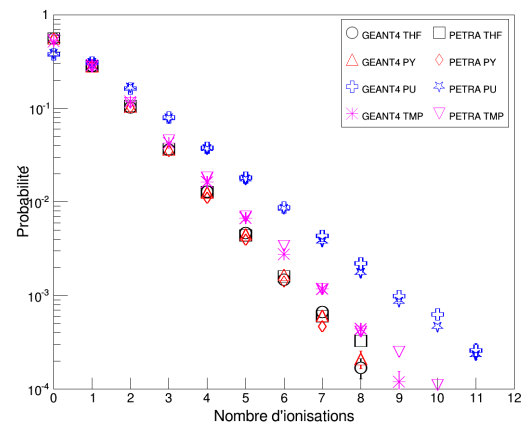
(a) 100 eV



(b) 250 eV



(c) 500 eV



(d) 1000 eV

FIGURE 3.1 – Distributions de probabilité du nombre d'ionisations occasionnées par un électron traversant un cylindre homogène constitué successivement de THF, TMP, PY et PU. Les distributions sont présentées pour des électrons de 100, 250, 500 et 1000 eV.

1.56 g/cm³.

$$d_{composite} = \frac{\sum_{k=1}^N (m_{mol_k} \times n_k)}{m_{cylindre}} \quad (3.4)$$

avec m_{mol_k} la masse totale des molécules du k^{th} composant, n_k le nombre de molécules du k^{th} composant (cf. le tableau 3.3) et $m_{cylindre}$ la masse totale du cylindre. Lors des si-

TABEAU 3.3 – Matériau composite représentatif de l'ADN.

Composants	Fraction massique	Masse molaire (g/mol)	Nb de molécules
Eau liquide	0.37	18.02	237
2-Désoxyribose	0.23	134.13	20
Phosphate	0.17	98.00	20
Cytosine	0.06	111.10	10
Thymine	0.07	126.11	10
Adénine	0.05	135.19	10
Guanine	0.05	151.13	10

mulations utilisées pour tester le système de management des matériaux, les ionisations ayant lieu dans le cylindre sont enregistrées et attribuées à l'électron primaire qui en est à l'origine. Cette fois, ces données sont utilisées pour calculer le nombre moyen d'ionisations M1 par particule primaire et par énergie, ainsi que la probabilité F2 d'obtenir plus de deux ionisations par événement. Les équations 3.5 et 3.6 illustrent comment ces deux valeurs sont calculées.

$$M_k = \sum_{k=1}^{\infty} (\nu_i \times k \times P_i) \quad (3.5)$$

avec $k=1$, ν_i un nombre d'ionisations and P_i la probabilité d'obtenir ν_i .

$$F_k = \sum_{i=k}^{\infty} (P_i) \quad (3.6)$$

avec $k=2$ et P_i la probabilité d'avoir i ionisations par particule primaire lancée. M1 et F2 sont des valeurs particulièrement intéressantes car ils représentent un lien direct entre le code de simulation MC et la nanodosimétrie expérimentale. En effet, en nanodosimétrie expérimentale, seules les ionisations peuvent être mesurées et M1 renseigne sur la complexité des clusters tandis que F2 peut être corrélé à la probabilité d'obtenir un nombre de DSB à l'aide de modèles spécifiques [BANTSAR, 2012; GARTY et al., 2010], certains modèles utilisent également ces grandeurs pour caractériser cliniquement les effets biologiques des rayonnements [ALEXANDER et al., 2015]. Les résultats obtenus pour le calcul de M1 et F2 avec Geant4.LDRI.01 et PTra sont présentés en figure 3.2. Notons que M1 et F2 sont donnés pour chacun des composants du matériau composite bien que la simulation ait eu lieu dans un matériau composite. Ceci est dû au fait que chaque interaction est attribuée à un composant du matériau composite au moment de calculer son impact sur la particule transportée (perte d'énergie, particule secondaire ...). Dans l'ensemble, les deux codes donnent des résultats similaires en terme de tendances avec une croissance jusqu'à environ 180 eV puis décroissance douce. Cependant des différences quantitatives existent et occasionnent, au maximum, un écart relatif de 25% entre les résultats calculés avec Geant4 et ceux obtenus avec PTra. Ces différences sont dues au fait que Geant4 et PTra utilisent des sections efficaces différentes pour l'eau liquide [BUG et al., 2012; VILLAGRASA et al., 2010] qui représente 37% de la masse du matériau composite utilisé. Bien

que ces différences existent et que des simulations basées sur les mêmes données pour l'eau liquide soient souhaitables, la vérification sera tout de même considérée comme effective dans ce travail en raison du bon accord global entre les résultats des deux codes : les tendances sont similaires et, malgré le fait que les écarts relatifs soient importants, ils ne dépassent pas 0.06 pour le M1 et 0.02 pour le F2 dans l'absolu.

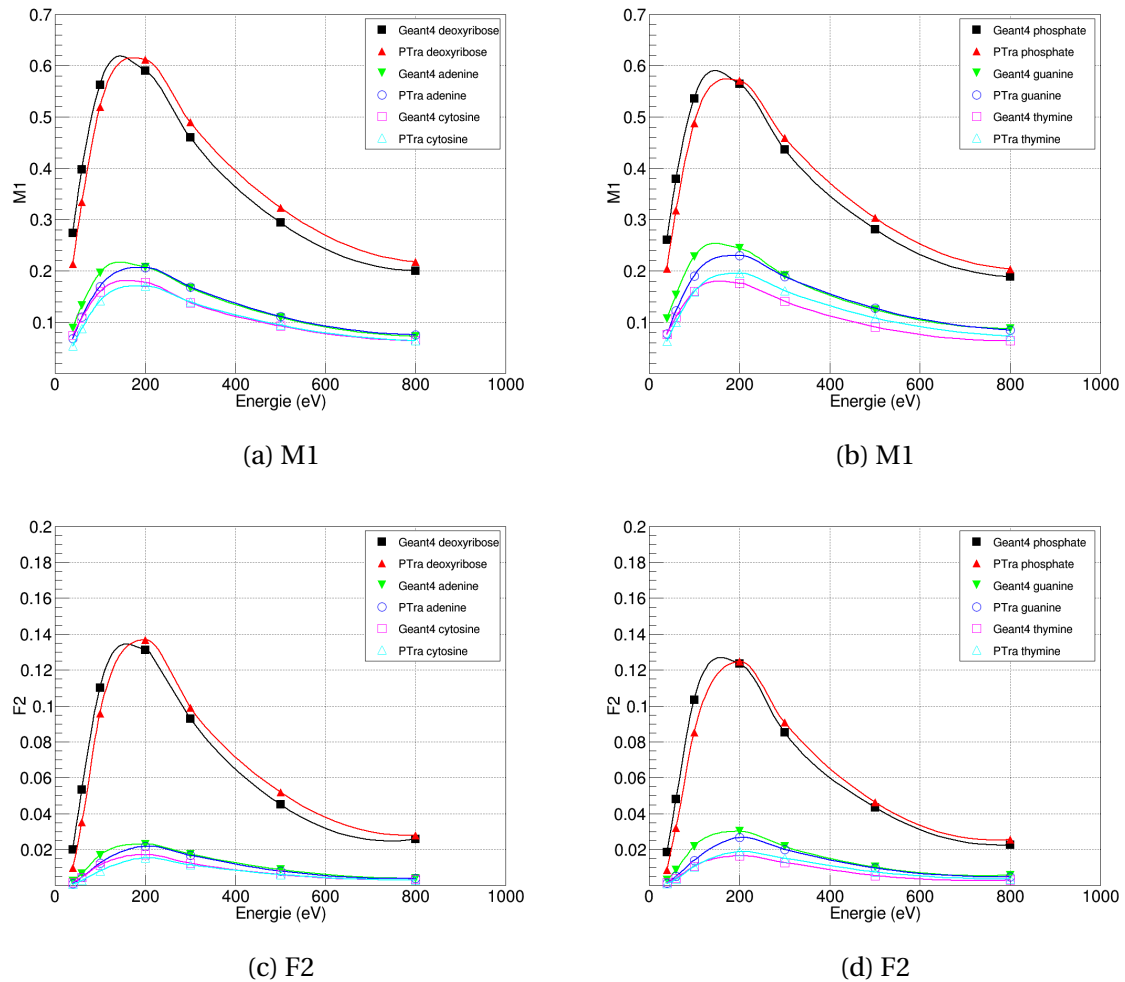


FIGURE 3.2 – M1 [(a) et (b)] et F2 [(c) et (d)] calculés avec Geant4 et PTra dans un matériau composite représentatif de l'ADN. Les simulations sont effectuées dans un cylindre traversé par des électrons et les ionisations engendrées par ces derniers sont enregistrées et associées à un des composants du matériau composite utilisé.

3.1.2 Choix d'une méthode pour l'attribution d'une interaction à un composant de l'ADN

Le matériau composite ADN précédemment mentionné peut-être utilisé dans une modélisation géométrique détaillée de l'ADN. La modélisation mise en œuvre dans ce travail décrit l'ADN de manière moléculaire comme indiqué en section 2.2.1. A ce propos, il est important de noter qu'il existe deux façons d'attribuer une ionisation, ou plus généralement une interaction physique, à un des constituants de l'ADN :

1. Du point de vue du transport des particules, chaque interaction est d'abord localisée spatialement à l'aide du libre parcours moyen puis, ensuite, un matériau est sé-

lectionné parmi les composants du matériau composite. Les caractéristiques de la particule transportée, les éventuelles particules secondaires générées et la quantité d'énergie déposée, sont alors déterminées en fonction du composant sélectionné. C'est ce composant qui est attribué en tant que matériau d'occurrence à chaque ionisation enregistrée durant la simulation. Au final et d'une certaine manière, cette sélection aléatoire d'un composant revient à attribuer un matériau d'occurrence à chaque ionisation simulée. Nous qualifierons ce type d'attribution de **matérielle** par la suite.

2. D'un point de vue purement spatial, toute interaction physique a lieu dans un volume géométrique. L'ADN est construit avec des nucléotides eux mêmes constitués à partir d'un agencement de six volumes distincts : 2-désoxyribose1, phosphate1, base1, base2, phosphate2, 2-désoxyribose2. Ainsi, chaque ionisation simulée dans l'ADN peut être attribuée à un volume spatial et c'est précisément ce qui est effectué en associant un volume d'occurrence aux ionisations enregistrées dans ce premier ensemble de simulations. Ce mode d'attribution sera qualifié de **volumique** par la suite.

Comme nous le verrons par la suite, ces deux méthodes d'attribution pouvant générer des résultats différents et il est nécessaire d'en choisir une lorsque le matériau composite ADN est utilisé. Pour guider ce choix, des calculs spécifiques ont été menés dans une géométrie constituée de 10 paires de nucléotides.

Configuration de la simulation

Afin de choisir une des deux méthodes d'attribution précédemment introduites, des calculs sont menés dans un morceau d'ADN constitué de 10 paires de nucléotides et représenté en figure 3.3. Ce morceau d'ADN est rempli avec le matériau composite ADN introduit en section 3.1.1. Des électrons mono-énergétiques sont utilisés en tant que particules primaires et leurs énergies cinétiques initiales vont de 50 à 800 eV, ce qui correspond approximativement aux énergies des électrons secondaires produits lors d'irradiations avec des protons ou ions alpha. La source des particules primaires se situe sur une surface cylindrique englobant les 10 paires de nucléotides et sont systématiquement envoyés en direction de ces dernières. En terme de résultats, les ionisations générées dans le cylindre sont enregistrées pour chaque électron primaire ainsi que pour chaque matériau et volume constitutif de l'ADN. Au final et à partir des ionisations enregistrées, le nombre moyen d'ionisations (M1) par électron primaire est obtenu pour chaque matériau et pour chaque volume.

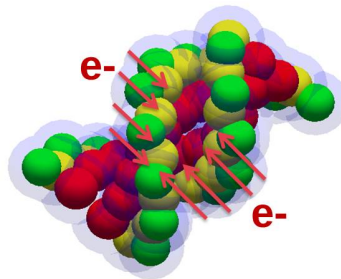


FIGURE 3.3 – Représentation d'un morceau d'ADN constitué de 10 paires de nucléotides rempli avec le matériau composite ADN. Les flèches rouges indiquent la direction de la source d'électrons. Cette source est composée d'électrons mono-énergétiques dont les énergies vont de 50 à 800 eV.

Nombres moyen d'ionisations obtenus avec l'attribution volumique et l'attribution matérielle

La figure 3.4 montre le nombre moyen d'ionisations (M1) calculé en fonction de l'énergie de l'électron primaire et du matériau attribué à chaque ionisation. Comme expliqué précédemment, cette attribution peut être effectuée de façon volumique ou matérielle au sein d'une géométrie de petite taille : 10 paires de nucléotides. C'est l'existence de deux modes d'attributions qui justifie la présence de deux courbes par composant du matériau composite ADN. Toutes les courbes de la figure 3.4 suivent une tendance identique composée d'une phase de croissance de 50 à 180 eV puis d'une décroissance douce de 200 à 800 eV. Cependant, cette tendance est plus ou moins prononcée selon le composant étudié et le mode d'attribution retenu. En effet, les phases de croissance et décroissance sont les plus importantes en amplitude pour la 2-désoxyribose (attribution matérielle et volumique) et le phosphate (attribution matérielle). Ce sont également pour ces deux composants que les courbes présentent les nombres d'ionisations les plus élevés pour toutes les énergies étudiées. Dans l'ensemble, les valeurs de M1 les plus élevées sont obtenues pour la désoxyribose avec une attribution "matérielle" des ionisations simulées. Les valeurs de M1 obtenues pour le phosphate, toujours avec une attribution "matérielle", sont légèrement inférieures. Les valeurs de M1 calculées pour l'adénine, cytosine, guanine et thymine sont relativement proches et systématiquement inférieures à celles obtenues pour la désoxyribose et le phosphate. Notons que la valeur de M1 calculée pour le phosphate à partir d'une attribution volumique est très similaire à celle obtenue pour la guanine avec une attribution matérielle des ionisations. Pour tous les composants de l'ADN, le fait de sélectionner une attribution matérielle augmente significativement le nombre moyen d'ionisations enregistrées par rapport à une attribution volumique.

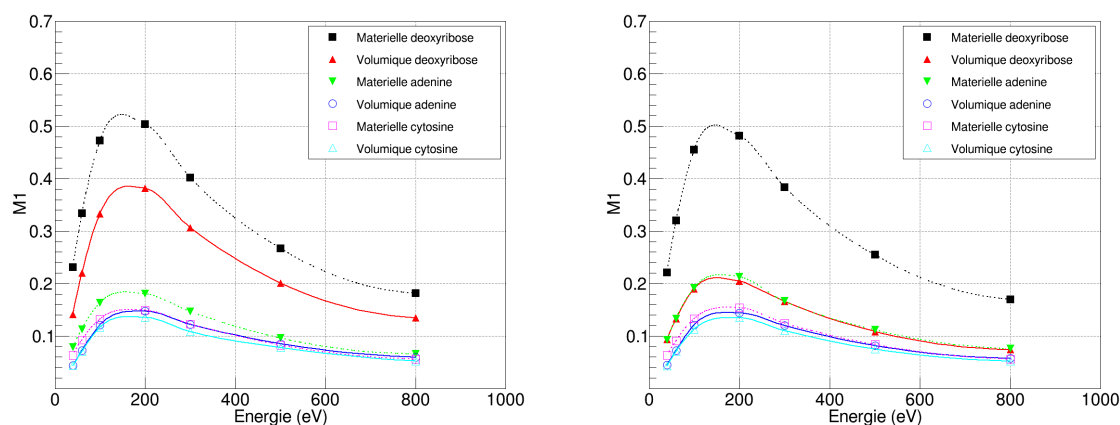


FIGURE 3.4 – Nombres moyens d'ionisations enregistrés dans 10 paires de nucléotides en fonction de l'énergie des électrons primaires, des composants de l'ADN et également du mode d'attribution des ionisations retenu : attribution "volumique" ou "matérielle". La masse volumique ρ du matériau composite ADN est de $1,56 \text{ g/cm}^3$.

Choix d'une méthode d'attribution

La figure 4.10 indique que les valeurs du nombre moyen d'ionisations attribuées à la désoxyribose ou au phosphate, avec une géométrie constituée de 10 paires de nucléotides, sont significativement différentes selon la méthode d'attribution des ionisations

utilisée : attribution matérielle ou attribution volumique. La manière d'attribuer un composant de l'ADN à une ionisation a donc une influence non négligeable et il est nécessaire d'en choisir une pour pouvoir déterminer la molécule concernée par une ionisation ou une interaction physique quelconque. Le cas de l'attribution matérielle d'une interaction conduit à des situations où, par exemple, une interaction est labellisée comme ayant eu lieu dans la désoxyribose alors que ses coordonnées spatiales peuvent correspondre au volume du phosphate. A l'inverse, l'attribution volumique implique l'existence de situations où, par exemple, une interaction est attribuée à la désoxyribose à partir de ces coordonnées spatiales alors que ses caractéristiques (énergie déposée par exemple) sont calculées à partir du matériau phosphate. Une façon de dépasser ces incohérences serait d'intégrer des considérations issues de la mécanique ou chimie quantique ainsi que suggéré dans la littérature [TERRISSOL et VRIGNEAUD, 2001]. L'intégration de ces considérations impliquerait de changer profondément la nature de la simulation effectuée et, notamment, de ne plus la baser sur la notion de section efficace classique. Toutefois, un tel travail dépasserait le cadre de cette thèse. Dans notre cas, il est intéressant de considérer la situation lorsque l'eau liquide est utilisée en tant qu'unique matériau. Dans cette configuration, le transport des particules a lieu dans l'eau liquide et la géométrie est prise en compte de manière conceptuellement séparée. Cette séparation est plus évidente avec d'autres simulations du domaine [BERNAL et al., 2013; FRIEDLAND et al., 2008; NIKJOO et GIRARD, 2012] dans lesquelles la géométrie représentant l'ADN est prise en compte *après* que le transport des particules a été effectué dans l'eau liquide. Au final, le fait d'utiliser ou pas la géométrie lors du transport des particules donne des résultats identiques tant que le matériau est le même partout. Lorsque l'eau liquide est le seul matériau, l'ensemble des simulations effectuées considèrent qu'une interaction physique a lieu dans un volume constitutif de l'ADN lorsque l'interaction est spatialement localisée dans ce dernier et le fait que les caractéristiques de l'interaction correspondent à celles de l'eau est négligé. Cela est équivalent à la méthode d'attribution volumique introduite dans ce travail. Finalement, c'est pour maintenir la cohérence vis à vis des simulations précédemment effectuées avec de l'eau liquide uniquement que **la méthode d'attribution volumique a été retenue** pour attribuer chaque interaction physique à un composant de l'ADN. Cette dernière sera utilisée pour les résultats présentés dans le chapitre 4.

3.1.3 Introduction du noyau cellulaire dans la simulation

Comme indiqué en section 2.1.3, un modèle géométrique de noyau cellulaire construit au sein de DnaFabric peut être exporté dans un fichier texte (.fab2g4dna) destiné à être importé dans une application utilisateur Geant4. Dans ce but, une classe nommée *PhysGeoImport* a été ajoutée à l'application *phys_geo* pour lui permettre de recréer un noyau cellulaire à l'aide des outils géométriques internes à Geant4, le tout à partir du fichier descriptif "nucleus_name.fab2g4dna" issue de DnaFabric. Cependant, le fichier "nucleus_name.fab2g4dna" décrit un noyau cellulaire rempli de voxels (cf. section 2.3.2) et ne fournit que leurs positions et noms respectifs. Ainsi, il est nécessaire de charger individuellement chacun des voxels utilisés dans le noyau cellulaire (cf. section 2.2.2) dans *phys_geo*. A cet effet, les voxels construits avec DnaFabric sont préalablement exportés dans des fichiers textes de type "voxel_name.fab2g4dna" pour être, ensuite, importés dans *phys_geo*.

L'import dans *phys_geo* des cinq voxels construits dans DnaFabric à partir de fichiers textes est un processus relativement long (quelques minutes) en raison de l'algorithme de découpage des volumes incorporé dans la classe *PhysGeoImport*. En effet, cet algorithme

recherche d'éventuelles intersections pour chacun des n volumes présents dans un voxel et, ce faisant, itère n^2 fois sur ces derniers. Par exemple, le voxel nommé "VoxelStraight" contient environ 5000 volumes distincts ce qui équivaut à 25 millions d'itérations. En pratique, certaines optimisations basées sur un découpage de l'espace sont incluses pour réduire le nombre d'itérations à effectuer mais ce dernier reste important et nécessite environ 2 minutes de calcul avec un ordinateur classique. L'algorithme de découpage utilisé est explicité par l'algorithme 3.1. Dans cet algorithme et d'un point de vue "Geant4", le découpage est effectué au moyen de deux opérations booléennes. Cela signifie que lorsqu'un chevauchement est détecté alors les deux volumes concernés seront "découpés" en leur soustrayant à chacun un volume dit "de découpe". Ce volume "de découpe" étant un pavé très large et très haut, le soustraire à un volume sphérique est équivalent à un découpage simple. Bien entendu, il est nécessaire de positionner correctement le volume "de découpe" au préalable. Le figure 3.5 illustre le concept de soustraction booléenne tel qu'il est utilisé dans le cadre de l'algorithme 3.1.

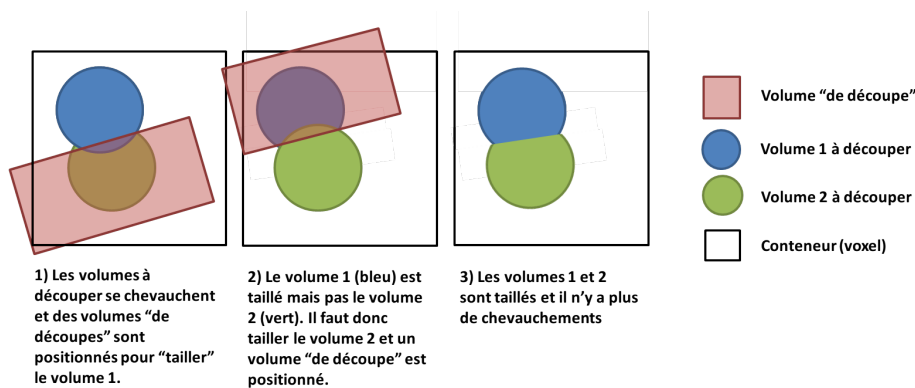


FIGURE 3.5 – Illustration du processus permettant d'éliminer les chevauchements implémenté dans phys_geo.

En plus du chargement des voxels et de l'élimination des chevauchements, le noyau cellulaire à proprement parler doit également être importé. Il s'agit de positionner environ 3 millions de voxels pour constituer un génome d'environ 6 milliards de paires de nucléotides. Cette étape de chargement dure environ 5 minutes et est particulièrement lourde en terme de mémoire vive (RAM). En effet, le chargement d'un noyau complet dans Geant4.LDRI.01 nécessite environ 4 Go de RAM. Bien qu'importante, cette quantité de mémoire vive nécessaire est toutefois raisonnable en raison de l'utilisation d'une optimisation propre à Geant4 lors du placement des voxels dans phys_geo : **la paramétrisation**. Cette optimisation consiste à fournir un ensemble de paramètres stockés dans un tableau de façon à ce que Geant4 les applique successivement à un volume lors du transport d'une particule. Généralement, le jeu de paramètres fourni est un ensemble de translations et de rotations ce qui permet de simuler la présence de nombreux volumes dans le calcul alors que Geant4 n'en stocke qu'un seul en mémoire. Présenté ainsi ce système est similaire à celui implémenté dans DnaFabric avec les "objets leurres" (cf. section 2.1.3). Dans Geant4, il est également possible de définir ce jeu de paramètres au moyen d'une équation ce qui diminue grandement l'espace mémoire requis mais rend la géométrie ainsi créée peu flexible (cf. les approches "équation" et "object" en section 2.1.3). Notons qu'une paramétrisation à l'aide d'équations se rapproche d'une technique utilisée en informatique graphique et connue sous le nom de **génération procédurale** [PARBERRY, 2014]. Dans le cas de l'import d'un noyau cellulaire et des voxels qu'il contient, les positions et orientations de ces derniers sont enregistrés dans un tableau qui

Données : Fichier ".fab2g4dna" ou "fichier d'import" contenant les positions de toutes les molécules à générer

Résultat : Chaque volume Geant4 est découpé de manière à éviter tout chevauchement

pour tous les placements effectués dans le fichier faire

Créer un volume sphérique dans Geant4 possédant les caractéristiques spécifiées dans le fichier d'import. Ce volume Geant4 est un G4Solid appelé "G4Orb". Identifier ce volume en tant que volume "de référence".

pour tous les placements effectués dans le fichier faire

Déterminer le volume résultant du placement sélectionné même si celui-ci n'est pas encore créé dans Geant4 et l'identifier en tant que volume "cible". Calculer la distance référence-cible et déterminer si ces deux volumes sphériques entrent en collision.

si collision détectée alors

Le volume de référence est découpé dans Geant4 au moyen de soustractions booléennes, ce qui signifie que la partie commune aux volumes de référence et de découpage est retirée du volume de référence. Le volume de découpage est un cube significativement plus grand que le volume de référence qui doit être positionné de manière à ce qu'une de ses faces coïncide avec le plan de découpe souhaité. Ce plan est systématiquement orthogonal à la droite reliant les centres des volumes de référence et cible. La position du plan de découpe le long de cette droite est calculée avec l'équation 3.7.

$$P_{inter} = \frac{R_{ref}^2 - R_{cib}^2 + d^2}{2 \times d} \quad (3.7)$$

où P_{inter} est la distance entre le centre du volume de référence et l'intersection du plan de découpe avec la droite reliant le volume de référence et le volume cible, R_{ref} est le rayon du volume de référence, R_{cib} est le rayon du volume cible, d est la distance de centre à centre entre le volume de référence et le volume cible. La position exacte du volume de découpe est obtenue avec l'équation 3.8.

$$P_{vol} = P - h \quad (3.8)$$

où P_{vol} est la position du volume de découpe et h la demi-taille du volume de découpe cubique.

Le volume de découpe est positionné en P_{vol} puis orienté de manière à faire coïncider une de ses faces avec le plan de découpe puis la soustraction booléenne suivante est effectuée :

$$V_{référencefinal} = V_{référenceinitial} - V_{découpe}$$

sinon

Passer au volume cible suivant.

fin

fin

retourner le volume de référence découpé.

fin

ALGORITHME 3.1 – Algorithme de découpage des sphères Geant4 pour éviter tout chevauchement

est ensuite fourni à Geant4 via une paramétrisation. La paramétrisation de cinq voxels distincts (les cinq types de voxel de la section 2.2.1) en une seule fois n'est pas pleinement supportée par Geant4 pour des raisons de gestion de mémoire (notamment en mode "multi-threading") et cela a rendu nécessaire quelques modifications de la classe *G4VPhysicalVolume*. L'activation de cette paramétrisation avec le mode MT a été mentionné en début de section en tant que spécificité de Geant4.LDRI.01.

3.1.4 Génération de données lors de la simulation

Les données produites

Les données générées par *phys_geo* peuvent être classées en deux catégories : les données descriptives de l'étape physique et les données qui seront utilisées en entrée de la simulation de l'étape physico-chimique. Les données sont toutes enregistrées dans le même fichier de sortie mais ce dernier est divisé en deux parties distinctes conformément à la catégorisation précédemment introduite. Précisons que le fichier de sortie est une n-tuple créée au moyen de la librairie ROOT-CERN [BRUN et RADEMAKERS, 1997].

L'enregistrement des données caractérisant l'étape physique soulève un problème de performance car la quantité de données potentiellement accessible est très importante. En effet, Geant4 offre, de part son aspect modulaire et open-source, la possibilité d'enregistrer n'importe quelle information utilisée durant la simulation. Précisons que si les informations les plus couramment utilisées sont très facile d'accès ce n'est pas forcément le cas des informations rarement demandées. Ainsi, l'énergie cinétique est accessible dans le *SteppingAction* via une simple fonction *GetKineticEnergy()* ce qui n'est pas forcément le cas du numéro d'identification attribué à un volume par exemple. Néanmoins, l'information est toujours accessible quel que soit le cas de figure. Il serait donc théoriquement possible d'enregistrer la totalité des informations liées aux aspects physiques de la simulation mais cela générerait une quantité très importante de données (une dizaine de Go pour quelques milliers de particules primaires) principalement due à l'utilisation de l'approche pas à pas. En effet, la majorité de ces données s'avèrent être les caractéristiques des interactions simulées lors des steps et ces derniers sont, comme expliqué en section 1.2.1, très nombreux dans le cadre d'une approche pas à pas. Pour réduire la quantité de données générées par *phys_geo*, le choix a été fait de n'enregistrer que les interactions ayant lieu dans l'ADN. Rappelons que l'ADN est constitué des volumes phosphate, 2-désoxyribose, adénine, guanine, cytosine, thymine et de leurs couches d'hydratation respectives (cf. section 2.2.1). Les informations enregistrées en sortie de la partie physique sont finalement :

- Le type d'interaction simulée qui peut être : élastique, ionisation, excitation (électronique ou vibrationnelle) ou encore attachement.
- L'énergie éventuellement déposée dans le milieu.
- Le nom de la particule impliquée.
- L'énergie cinétique pré et post interaction de la particule.
- Le numéro de l'événement en cours, c'est à dire le numéro associée à la particule primaire responsable directement ou indirectement de l'interaction simulée.
- Le nom du volume dans lequel l'interaction a lieu.
- L'identifiant de la paire de nucléotide associée au volume d'occurrence de l'interaction. Cet identifiant est un nombre entier contenu dans une variable nommée

CopyNumber et qui permet d'identifier précisément un volume à condition, toutefois, d'être utilisé conjointement avec le nom de ce dernier et un numéro de voxel. La figure 3.6 illustre l'usage combiné de l'identifiant et du nom pour repérer un volume précis.

- Le nom du voxel dans lequel se trouve le volume d'occurrence de l'interaction.
- L'identifiant du voxel.
- La position de l'interaction (x, y, z) dans le référentiel du voxel correspondant.

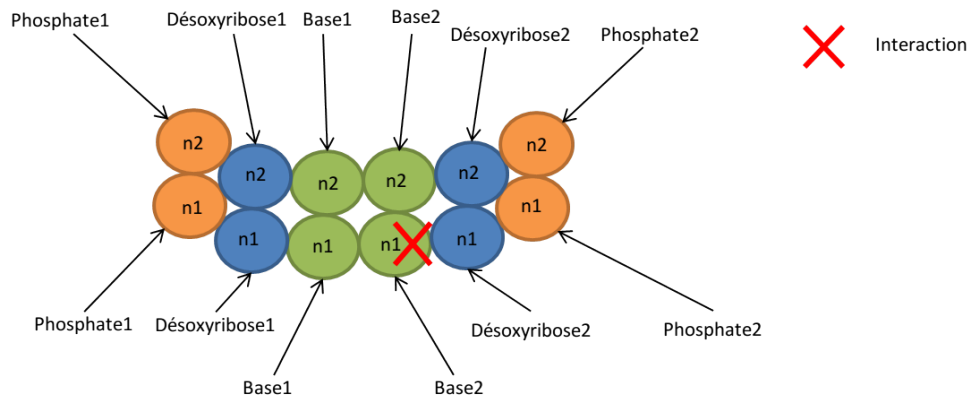


FIGURE 3.6 – Identification d'un volume avec son nom et l'identifiant de la paire de nucléotides. Deux paires de nucléotides sont représentées et leurs identifiants associés sont les numéros 1 et 2. Les noms des volumes sont indiqués avec les flèches noires. Une interaction a lieu à l'endroit repéré par la croix rouge. Le volume d'occurrence de l'interaction est caractérisé par son nom "Base2" et l'identifiant "1" du nucléotide.

Les données requises pour la simulation de l'étape physico-chimique sont, en principe, les positions des électrons solvatés et des molécules d'eau ionisées ou excitées à l'échelle du noyau cellulaire entier. Ceci représentant, encore une fois, une quantité très importante de données à l'échelle du noyau entier (plusieurs Go) la décision a été prise de limiter l'enregistrement de ces données aux électrons solvatés et molécules d'eau ionisées ou excitées situés **dans un voxel**. Cela revient à effectuer une approximation en considérant que les espèces chimiques générées en dehors des voxels sont trop éloignées de l'ADN pour l'atteindre et influencer le résultat attendu (dommages à l'ADN). Les voxels ayant une largeur de 50 nm et la fibre ADN un diamètre d'environ 30 nm, il existe un espace de 10 nm en moyenne entre l'ADN et la paroi du voxel ce qui est considéré comme suffisant vis à vis de la distance de diffusion moyenne du radical OH^\bullet . Cette distance de diffusion dépend de nombreux facteurs mais est généralement estimée comme étant inférieure à 10 nm [ABOLFATH et al., 2013; REZAEI et al., 2013]. Finalement, les informations enregistrées pour la simulation de l'étape physico-chimique sont :

- Le type de l'élément à transmettre à l'étape physico-chimique : électron solvaté ou molécule d'eau ionisée ou encore excitée. Dans le cas d'une molécule d'eau ionisée ou excitée, des informations concernant son état d'excitation sont également fournies.
- Le numéro de l'événement en cours qui correspond au numéro de la particule primaire directement ou indirectement à l'origine de la création de l'électron solvaté ou de la molécule d'eau instable.
- Le nom du volume dans lequel se trouve l'élément à transmettre. Ce nom est toujours celui d'un des cinq voxels introduits dans la simulation car les électrons solvatés et molécules d'eau instables générées hors des voxels sont ignorés.

- L'identifiant du voxel (*CopyNumber*).
- La position relative de l'élément dans le voxel.

Identification des dommages directs

L'identification de dommages directs à l'ADN à partir de l'ensemble des données issues de la simulation de l'étape physique n'est pas immédiate. La cartographie des interactions ayant eu lieu dans l'ADN est, en effet, directement disponible mais elle ne correspond pas à la cartographie des dommages directs à l'ADN (SSB) car un dommage n'est pas forcément équivalent à une cassure de l'ADN pour plusieurs raisons. D'abord, toutes les interactions ne peuvent endommager l'ADN. Par exemple, les interactions élastiques ont, par définition, un faible impact sur le milieu ambiant et donc sur l'ADN. Ainsi, il est légitime de considérer que les interactions élastiques ne peuvent occasionner de dommage. Ensuite, un dommage à l'ADN peut être différent selon le volume touché par une interaction : base, 2-désoxyribose, phosphate, couche d'hydratation... Enfin, si des interactions peuvent effectivement endommager l'ADN, le lien de causalité n'est pas forcément direct et d'autres facteurs entrent en jeu. L'identification de la chaîne d'événements physiques menant à un dommage à l'ADN est un champ de recherche actif dans lequel il est généralement considéré que les ionisations et excitations des molécules d'ADN peuvent engendrer des dommages dans certaines conditions [GOODHEAD, 1994]. Notons qu'une ionisation localisée dans la couche d'hydratation entourant l'ADN peut causer un dommage via un processus de transfert de charge [SWARTS et al., 2007]. Précisons également qu'un processus d'attachement dissociatif [CELIBERTO et al., 2012; LAPORTA et al., 2015] impliquant directement l'ADN peut causer des dommages via un effet de résonance lié à la structure de l'ADN [BALD et al., 2012; BOUDAIFFA et al., 2000]. Ceci implique notamment que des électrons avec des énergies inférieures à celles nécessaires pour exciter ou ioniser l'ADN peuvent tout de même occasionner des dommages. Il s'agit d'électrons que nous qualifierons de "pré-solvatés". L'absence de caractérisation précise et globale du processus de création d'un dommage direct à l'ADN conduit à effectuer des approximations au sein d'un modèle. Le modèle généralement retenu dans les simulations mécanistiques est une sélection basée sur l'énergie déposée dans certains constituants de l'ADN. Les caractéristiques de cette sélection varient en fonction des codes de calculs et des auteurs [FRIEDLAND et al., 2003; NIKJOO et al., 1997].

Dans FullSim, un dommage direct à l'ADN est identifié par défaut comme une succession d'ionisations ou excitations dans le phosphate ou la 2-désoxyribose (couche d'hydratation incluse) ayant déposé une énergie supérieure à **17,5 eV au total** [NIKJOO et al., 2001]. Alternativement, une méthode basée sur une probabilité d'acceptation linéaire est également implémentée. Dans ce cas, lorsque qu'une énergie supérieure à **37,5 eV** est déposée dans le squelette de l'ADN alors un dommage direct est comptabilisé. Si l'énergie déposée est inférieure à 37,5 eV mais de même supérieure à 5 eV, alors un tirage aléatoire est effectué pour déterminer la présence d'un dommage. La probabilité d'acceptation est alors linéaire et calculée avec l'équation 3.9.

$$P_{acceptation} = \frac{1}{37.5 - 5} \cdot E_{déposée} + \frac{5}{37.5 - 5} \quad (3.9)$$

avec $P_{acceptation}$ la probabilité d'acceptation et $E_{déposée}$ l'énergie déposée dans le nucléotide (les volumes des bases ne sont pas pris en compte pour enregistrer l'énergie déposée). Cette méthode est également utilisée dans le code PARTRAC [FRIEDLAND et al., 2003] et son incorporation dans FullSim permet une comparaison directe avec les données de la

littérature issues de ce code. Comme nous le verrons en section 3.3, ces différentes méthodes de sélection des dommages directs peuvent être modifiées à des fins d'analyse.

3.2 Simulation des étapes physico-chimique et chimique

La simulation des étapes physico-chimique et chimique s'effectue dans une même application utilisateur nommée **chem_geo**. Cette application s'appuie sur Geant4.LDRI.01 et reçoit en tant que paramètres (i.e. "arguments") la description du noyau ainsi que les résultats issus de la simulation de l'étape physique (**phys_geo**). Tout comme **phys_geo**, **chem_geo** est construite sur le modèle évoqué en section 1.3.3 et a pour particularité de faire fonctionner le module "chimie" de Geant4-DNA avec une description géométrique de l'ADN. La simulation contient différents paramètres ajustés de manière à construire une modélisation cohérente des étapes physico-chimique et chimique : temps de simulation, durée des pas en temps, taux de réactions ADN-radicaux... La prise en compte d'un modèle géométrique de l'ADN au sein du module "chimie" de Geant4-DNA, les données générées par **chem_geo** et les différents paramètres ajustés constituent les sous-parties de cette section.

3.2.1 Prise en compte du modèle géométrique

La prise en compte du noyau cellulaire dans le module "chimie" de Geant4-DNA n'est pas nativement supportée par Geant4-DNA. D'abord parce que le noyau est décrit avec plusieurs fichiers textes (noyau et voxels) générés par DnaFabric et, ensuite, parce que le module "chimie" de Geant4-DNA est conçu pour simuler la radiolyse de l'eau dans un milieu homogène sans géométrie. Les développements conjointement effectués dans **chem_geo** et Geant4.LDRI.01 ont permis de dépasser ces limitations initiales et sont exposés ci-dessous. Notons qu'à partir de cette partie, le terme "molécule d'ADN" sera utilisé pour désigner ce qui était précédemment nommé "volume d'ADN" ou encore "constituant de l'ADN". Nous dirons donc qu'une paire de nucléotides est constituée de six molécules : phosphate 1, 2-désoxyribose 1, base 1, base 2, phosphate 2 et 2-désoxyribose 2.

Découpage en simulations intermédiaires et indépendantes

Suite à la simulation de l'étape physique, les nombreuses données obtenues sont toutes enregistrées dans le fichier de sortie décrit en section 3.1.4. Ce fichier est fourni à un programme d'analyse qui se charge de passer en revue les données spécifiquement dédiées à la partie physico-chimique et de les classer par événement (ou event) et par voxel. Ce classement est ensuite utilisé pour exécuter l'application **chem_geo** indépendamment pour chaque binôme "event/voxel" présent dans les données. En d'autres termes, cela signifie que la simulation des étapes physico-chimique et chimique est fragmentée en de multiples simulations intermédiaires et indépendantes. Chacune de ces simulations correspondant à un binôme event/voxel tel qu'illustré en figure 3.7. Les résultats de ces simulations intermédiaires sont ensuite rassemblés au moyen d'un second programme d'analyse. Cette fragmentation de la simulation introduit cependant une approximation car les espèces chimiques diffusant à l'intérieur d'un voxel ne peuvent en sortir pour interagir avec celles des voxels environnant. Les simulations chimiques effectuées au niveau des voxels sont donc isolées et les interactions inter-voxels négligées. Néanmoins, le découpage des étapes physico-chimique et chimique en plusieurs simulations intermédiaires induit une série d'avantages en terme de mémoire, rapidité et facilité de cal-

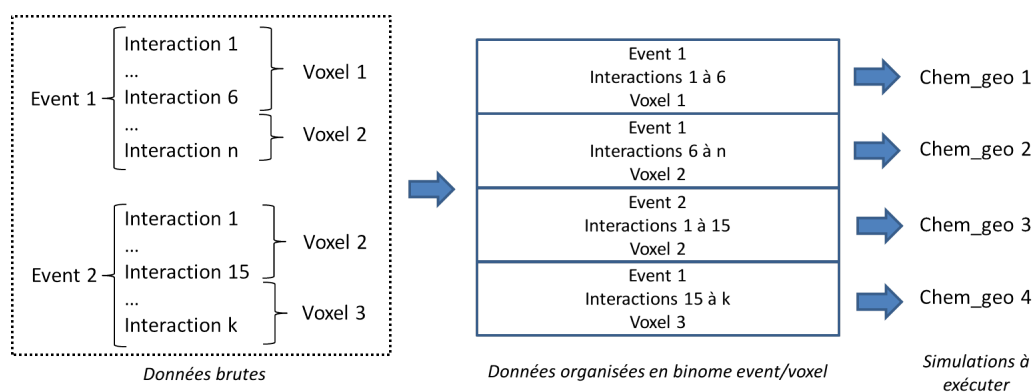


FIGURE 3.7 – Organisation des données issues de la simulation de l'étape physique ("données brutes") en binôme event/voxel. Chaque binôme correspond à l'exécution d'une "simulation intermédiaire" indépendante des étapes physico-chimique et chimique. Les "interactions" correspondent aux molécules d'eau ionisées (ionisation) ou excitées (excitation) et aux électrons solvatés (solvatation).

cul. En effet, chaque sous-simulation prise individuellement ne représente qu'un voxel et donc une quantité de molécule d'ADN limitée. Cette limitation du nombre de molécules d'ADN est drastique vis-à-vis de la simulation d'un noyau entier : environ 36 milliards de molécules d'ADN à l'échelle du noyau pour seulement quelques milliers par voxel. De plus, la prise en compte dans la simulation de molécules d'ADN très éloignées des interactions physique (i.e. de la trace) ne présente pas d'intérêt car ces dernières ont une probabilité négligeable de rencontrer un radical et donc d'être endommagées. Les simulations intermédiaires sont également bien plus rapides qu'une seule simulations globale ce qui rend certains résultats préliminaires accessibles avant la véritable fin du calcul. La fin du calcul ayant lieu lorsque des simulations intermédiaires listées ont été exécutées. Dans le même registre, si le calcul doit être interrompu pour une raison quelconque alors seuls les résultats de la simulation intermédiaire brutalement stoppée sont perdus. En effet, les simulations intermédiaires précédemment exécutées ont enregistré leurs résultats dans des fichiers de sortie dédiés complètement indépendants de la simulation stoppée. La lisibilité des calculs est également améliorée car les fichiers d'entrée et de sortie associés à chaque simulation intermédiaire sont de tailles modestes (quelques centaines de lignes) et donc lisibles à l'œil nu. Ceci permet de vérifier aisément leur cohérence et, éventuellement, d'y apporter des modifications. Enfin, le fait d'avoir une série de simulations intermédiaires à exécuter indépendamment les unes des autres constitue la configuration idéale pour l'utilisation du **pseudo-parallélisme**. C'est à dire le lancement simultané des simulations sur différentes unités de calcul (threads). Dans cette configuration, l'utilisation du pseudo-parallélisme induit un gain en temps directement proportionnel au nombre d'unités de calcul utilisées. Par exemple, le fait de répartir les simulations intermédiaires sur 10 unités de calcul divisera par 10 le temps de calcul comparé à une configuration séquentielle où les simulations intermédiaires sont lancées les unes à la suite des autres. Notons que le pseudo-parallélisme est à distinguer du véritable parallélisme qui consiste à exécuter un programme spécialement "parallélisé". Ce programme s'exécute sur une unique unité de calcul dans un premier temps et réparti de lui même les tâches à exécuter sur les différentes unités de calcul mises à sa disposition. Geant4 permet un tel parallélisme mais uniquement pour la simulation de l'étape physique. L'application utilisateur phys_geo en fait d'ailleurs usage.

Introduire l'ADN dans une simulation basée sur le module "chimie" de Geant4-DNA

Le module "chimie" de Geant4-DNA (version 10.01) est conçu pour simuler la diffusion et les réactions d'espèces chimiques dans un milieu homogène (cf. section 1.3). Cela signifie, dans le cadre de Geant4-DNA, que les espèces chimiques se déplacent toutes lors de chaque pas en temps et qu'elles ne perçoivent pas les volumes géométriques définis dans Geant4. Dès lors, l'introduction d'une représentation de l'ADN requiert des modifications dans le module "chimie" de Geant4-DNA pour y ajouter le concept d'objet chimique statique. Principalement deux types de modifications sont envisageables pour atteindre cet objectif :

1. Intégrer dans le module "chimie" la gestion des volumes classiquement définis dans Geant4.
2. S'appuyer sur le concept de molécule déjà existant et définir des molécules non-diffusantes, c'est à dire statiques .

Dans ce travail, c'est la solution 2 qui a été retenue. En effet, cette solution a pour avantages de s'appuyer sur des composants et une architecture déjà présents dans le module "chimie" tout en conservant une certaine cohérence vis-à-vis des concepts sous-jacents. En effet, la séparation entre les volumes classiques de Geant4 et les molécules du module "chimie" se justifie vis-à-vis des comportements très différents de ces deux entités : les volumes Geant4 sont statiques et ne réagissent pas entre eux tandis que les molécules sont dynamiques et réagissent entre elles. Toutefois, il est intéressant de remarquer que la solution 1 pourrait déboucher sur un nouveau concept unifiant les caractéristiques de ces deux entités. La mise en œuvre d'un tel concept nécessiterait de nombreuses modifications touchant à divers composants de Geant4 (Navigation, Géométrie, Matériaux...) en raison, précisément, du fait que les caractéristiques à unifier soient différentes et actuellement gérées par des composants distincts.

Techniquement, la définition de molécules statiques dans le module "chimie" implique d'attribuer aux molécules en question un coefficient de diffusion nul. Avec le code la version 10.01, cela les rend effectivement immobiles du point de vue de la diffusion. Toutefois, des réactions chimiques entre les molécules statiques de l'ADN et les espèces chimiques dynamiques doivent être intégrées dans la simulation et, parmi ces réactions, certaines seront du type :



avec *histone* une "molécule" représentant le complexe de protéines histones et *molécule* une espèce chimique quelconque. Dans un tel cas de figure, la molécule *histone* est à la fois un réactif et un produit de la réaction et, parfois, il peut être souhaitable de ne pas modifier sa position suite à la réaction. Hors, le module "chimie" ne prend pas en charge un tel comportement dans sa version de base où les produits de réaction sont automatiquement positionnés près du lieu de réaction. L'ajout de ce comportement a donc été effectué en implémentant une nouvelle "sorte" de réactions possédant une liste de réactifs/produits "fixes" que nous qualifierons de **réactions conservatrices**. Ces réactions sont définies directement dans l'application utilisateur en charge de la simulation.

L'ajout des molécules statiques et des réactions conservatrices dans le module "chimie" rend la prise en compte de l'ADN défini dans DnaFabric possible. Cette dernière est effectuée par une classe dédiée appartenant à l'application `chem_geo` : `ChemGeoImport`. `ChemGeoImport` lit les fichiers textes (.fab2g4dna) exportés par DnaFabric et qui décrivent les cinq voxels utilisés dans ce travail (cf. section 2.2.2). Une fois les fichiers lus, les volumes constituant l'ADN en sont extraits puis convertis en molécules. Dans ce

but, les constituants de l'ADN sont préalablement définis, au sein de chem_geo, en tant que molécules statiques capables, éventuellement, de réagir avec certaines espèces chimiques. En pratique, un volume nommé "Adénine" sera converti en une molécule statique du même nom, ensuite insérée dans la simulation au moyen du *MolecularGun* (cf. section 1.3.3). Le tableau 3.4 liste les réactions entre l'ADN et les espèces chimiques qui sont prises en compte dans chem_geo. Dans ce travail, la totalité des réactions impliquant les molécules de la double hélice de l'ADN ont lieu avec le radical OH• et induisent la disparition de la simulation des réactifs impliqués. Seules les réactions impliquant le radical OH• sont prises en compte car ces réactions sont reconnues comme particulièrement critiques vis à vis des dommages à l'ADN [MILLIGAN et al., 1996; ROOTS et OKADA, 1975]. La réaction du tableau 3.4 qui implique le complexe de protéines histones ne provient pas de la littérature. En effet, il s'agit une nouvelle fois d'un parti pris spécifique à la simulation de l'étape chimique et qui consiste à donner au complexe de protéines histones un rôle de "scavenger" (absorbeur). En pratique, ce complexe est représenté par une molécule sphérique ("histone") qui absorbe toute molécule entrant en contact avec elle. Cette absorption correspond à une réaction conservatrice où le complexe de protéines est à la fois un réactif et un produit.

TABLEAU 3.4 – Les réactions implémentées dans chem_geo entre l'ADN et les espèces chimiques [BUXTON et al., 1988; KREIPL et al., 2009]. La dernière réaction ne provient pas de la littérature et est spécifique à chem_geo où l'histone a un rôle de "scavenger" absorbant toutes les molécules entrant en contact avec elle. Cette réaction se déclenchant lors d'un contact direct, elle ne possède pas de taux de réaction.

Réaction	Taux de réaction ($10^{-3} m^3 / (mol \cdot s)$)
2 – désoxyribose + OH•	$2.5 \cdot 10^9$
Adénine + OH•	$6.10 \cdot 10^9$
Guanine + OH•	$9.20 \cdot 10^9$
Thymine + OH•	$6.40 \cdot 10^9$
Cytosine + OH•	$6.10 \cdot 10^9$
Histone + molécule → histone	x

Les électrons solvatés ainsi que les molécules d'eau ionisées ou excitées durant l'étape physique sont ajoutés à la simulation via le *MolecularGun* en plus des molécules de l'ADN. Leurs nombres et leurs positions de départ sont déterminés à partir du fichier texte créé suite à l'analyse des résultats de phys_geo (cf. les binômes event/voxel). Ce fichier contient également la liste des molécules d'ADN préalablement ionisées ou excitées durant la simulation de l'étape physique pour qu'elles soient retirées de la simulation des étapes physico-chimique et chimique. En effet, ces molécules d'ADN ne sont plus considérées comme susceptibles de réagir avec le radical OH• car leur structure a été modifiée par l'interaction physique subie. Il s'agit, encore une fois, d'un parti pris propre à la simulation.

Au final, chaque exécution de chem_geo se déroule en cinq temps distincts :

1. Chargement et insertion des molécules d'ADN décrite à l'aide du fichier ".fab2g4dna".
2. Lecture du fichier texte, produit par la simulation de l'étape physique, listant les électrons solvatés, les molécules d'eau instables et les molécule d'ADN déjà ionisées. Les électrons et molécules d'eau sont insérés dans la simulation tandis que les molécules d'ADN ionisées lors de l'étape physique en sont retirées.
3. Lancement de la simulation.

4. Simulation de l'étape physico-chimique : dissociation des molécules d'eau instables.
5. Simulation de l'étape chimique en incluant les réactions entre le radical OH[•] et l'ADN.

3.2.2 Génération de données lors de la simulation

Données produites

Les données générées lors de chaque exécution de `chem_geo` sont centrées autour de la notion de réaction chimique. Comme indiqué en section 1.3.3, la méthode `UserReactionAction` de la classe `TimeStepAction` est appelée à chaque fois qu'une réaction se produit lors de la simulation. C'est également cette méthode qui remplit le fichier de sortie associé à la simulation en cours. Ce fichier est, encore une fois, un ntuple généré à l'aide de la librairie ROOT-CERN. Finalement, ce fichier contient les informations suivantes :

- Noms des réactifs et produits.
- Positions des réactifs et produits dans le voxel considéré.
- Temps auquel la réaction a lieu. Il s'agit du temps simulé et non pas du temps réellement écoulé durant la simulation.

Des informations complémentaires sont ajoutées si l'un des réactifs est une molécule d'ADN :

- Numéro du brin d'ADN également appelé "strand" : 1 ou 2.
- Identifiant de la paire de nucléotides associée à la molécule d'ADN impliquée dans la réaction (*CopyNumber*).

Le nom du fichier dans lequel les données sont stockées contient les informations requises pour caractériser le binôme event/voxel à l'origine de l'exécution de `chem_geo`, soit le numéro de l'événement ainsi que le nom et l'identifiant du voxel concerné.

La quantité de données produites lors de chaque exécution de `chem_geo` occupe une place mémoire restreinte mais de très nombreuses exécutions sont requises pour chaque particule primaire lancée durant la simulation de l'étape chimique. A titre indicatif, le lancement de 1000 protons de 1 MeV, en tant que particules primaires, au sein de `phys_geo` génère environ 45000 binômes event/voxel avec un noyau de cellule fibroblaste. Cela signifie qu'il y a 45000 simulations `chem_geo` à exécuter. Sachant qu'une simulation `chem_geo` génère environ 46 Mo de données, cela représente un total de 2 Go de données pour l'ensemble des 45000 simulations, ce qui est conséquent.

Identification des dommages à l'ADN

L'identification des dommages à l'ADN dans `chem_geo` est une sélection effectuée sur l'ensemble des réactions impliquant le radical OH[•] et l'ADN. Ces réactions sont toutes enregistrées dans un fichier de sortie qui peut être analysé de diverses manières. L'analyse présentée dans ce paragraphe est celle appliquée par défaut dans FullSim et dans laquelle seules les réactions entre le radical OH[•] et la 2-désoxyribose sont extraites du fichier de sortie. Une fois ces réactions extraites, un tirage aléatoire est utilisé pour n'en sélectionner que 42%. Cette probabilité est une manière de prendre en compte les cinq sites de la molécule de 2-désoxyribose avec lesquels le radical OH[•] peut réagir [AYDOGAN et al., 2002; BALASUBRAMANIAN et al., 1998]. En effet, le taux de réaction associé à la réaction 2 – désoxyribose + OH[•] dans `chem_geo` prend en compte l'ensemble des sites réactifs

de la 2-désoxyribose alors que, dans notre cas, certains sites sont peu accessibles en raison de la structure de l'ADN. Il semblerait que, dans de l'ADN, deux sites de la molécule de 2-désoxyribose soient particulièrement accessibles justifiant ainsi l'application d'une probabilité de 42% ($\approx \frac{2}{5}$) [BALASUBRAMANIAN et al., 1998].

3.2.3 Quelques paramètres spécifiques à l'étape chimique

Le développement de chem_geo a nécessité des choix concernant certains aspects de la simulation et ces derniers constituent des parti-pris intégrés à la modélisation. Les raisons à l'origine de ces partis pris sont diverses : absence de données dans la littérature, données inadaptées au contexte de la simulation (expérience, type de cellule...), impossibilité d'intégrer des données dans la simulation pour des raisons liées au calcul (performance, mémoire...), etc. Un certain nombre d'entre eux, les plus significatifs, sont énoncés dans cette section avec leurs valeurs par défaut dans chem_geo. Ces valeurs par défaut ont été ajustées de façon à retrouver les résultats expérimentaux du chapitre 4 [FRIEDLAND et al., 2002].

Le temps

Le temps total simulé est un paramètre important de chem_geo car il conditionne le temps pendant lequel les espèces chimiques issues de la radiolyse de l'eau, notamment le radical OH^\bullet , peuvent diffuser dans le milieu et interagir avec l'ADN. Si ce temps est trop court alors ces dernières ne réagissent pas mais s'il est trop long alors elles réagiront trop. En effet, une telle limite de temps n'existe pas en conditions expérimentales mais, dans ces conditions, la cellule contient de nombreux éléments interagissant avec les espèces chimiques issues de la radiolyse de l'eau. L'ensemble de ces interactions diminue significativement leur nombre et il s'agit d'un processus connu en tant que "scavenging". Cependant, chem_geo ne simule pas directement ces éléments "scavengers" contenus par la cellule (O_2 , protéines..) et doit donc les prendre en compte de manière indirecte. Ceci est effectué en limitant le temps de simulation à la durée de vie moyenne du radical OH^\bullet estimée à quelques nanosecondes dans notre cas [ABOLFATH et al., 2013]. En accord avec cela, le temps total simulé dans chem_geo est fixé à 2,5 ns par défaut. L'influence de ce paramètre est étudiée plus en détail dans le chapitre 4.

L'histone

Le complexe de protéines histones est représenté par une sphère dans chem_geo et, comme mentionné en section 3.2.1, toute espèce chimique entrant en contact avec cette sphère est retirée de la simulation. L'histone a donc un effet similaire à celui d'un "scavenger" qui vient s'ajouter à celui de la durée limitée de la simulation. L'absorption ayant lieu lors du contact, cet effet est directement lié à la taille de la sphère représentative du complexe. Dans chem_geo, la sphère a une taille de 2,4 nm par défaut et, comme dans le cas du temps total de simulation, l'influence de ce paramètre est étudiée dans le chapitre 4.

Le découpage en voxel

La simulation de l'étape chimique se déroule dans des voxels isolés ce qui signifie que les espèces chimiques ne peuvent passer d'un voxel à un autre. Cette restriction spatiale

limite les réactions possibles et, ainsi, perturbe l'équilibre de la simulation. L'ampleur de cette perturbation est étudiée dans le chapitre 4.

Phosphate

Une paire de nucléotides est constituée de six molécules introduite dans la simulation. Cependant, `chem_geo` ne simule aucune réaction impliquant le radical OH^\bullet et les deux molécules de phosphate situées aux extrémités de la paire de nucléotides. Si aucune espèce chimique ne peut réagir avec les molécules de phosphate, cela signifie qu'elles sont absolument transparentes du point de vue de `chem_geo`. En pratique, un radical OH^\bullet pourra réagir avec une molécule de désoxyribose en empruntant un chemin sur lequel se trouve une molécule de phosphate et la traverser comme si elle n'était pas présente. Cette "transparence" revient à négliger l'effet protecteur de la molécule de phosphate augmentant ainsi l'accessibilité de la 2-désoxyribose.

3.3 Organisation de la simulation des trois étapes

Comme indiqué en début de chapitre, les outils permettant de simuler les étapes physique, physico-chimique et chimique sont rassemblés et organisés dans la chaîne de calcul FullSim. FullSim est ainsi constituée des simulations `phys_geo` et `chem_geo` mais, également, des programmes chargés d'analyser les données produites et des scripts responsables de l'exécution des différents éléments de la simulation. Le lancement de ces éléments en se passant de la chaîne de calcul est possible mais fort peu pratique. Par exemple, la simulation de l'étape chimique requiert d'exécuter plusieurs milliers de fois l'application utilisateur `chem_geo` avec un jeu de paramètres précis (couple event/voxel) pour chaque exécution. Dans un premier temps, cette section présente le contenu et l'architecture de la chaîne de calcul FullSim et, dans un second temps, détaille son utilisation et l'ordre dans lequel ses éléments sont activés.

3.3.1 Contenu et architecture

Le lancement d'une simulation avec FullSim implique l'exécution d'une dizaine de scripts et programmes :

- L'application utilisateur **phys_geo** (C++, Geant4-DNA, version LDRI.01).
- Le programme **extractChemInput** (C++) qui analyse les résultats de `phys_geo` pour en extraire les données nécessaires au lancement de `chem_geo`.
- L'application utilisateur **chem_geo** (C++, Geant4-DNA, version LDRI.01).
- Le programme **extractSSB** (C++) qui analyse les résultats de `phys_geo` et `chem_geo` pour détecter les dommages subis par l'ADN.
- Le programme **DBScan** (Fortran) qui détecte des clusters de dommages à l'ADN à partir des données fournies par `extractSSB`. Par défaut, ce programme est configuré pour détecter des clusters de dommages à l'ADN.
- Le programme **postProcessDBScan** (C++) se charge d'analyser les résultats bruts de `DBScan` et de les convertir en cassures simple ou double brin (SSB ou DSB). Ces dernières sont disponibles en valeur absolue pour le noyau entier et par chromosome. Les valeurs moyennes suivantes sont également calculées par défaut : DSBs/event et SSBs/event.

- Le programme **statControl** (C++) contrôle la pertinence statistique du nombre de DSBs issu de PostProcessDBScan. StatControl écrit ce nombre ainsi que l'incertitude statistique qui y est associée dans un fichier de sortie nommé "control_stat.txt".

Ces programmes ne sont pas conçus pour être exécutés directement en raison des nombreux paramètres qu'il est nécessaires de leur fournir. La gestion de ces paramètres ainsi que celle de l'exécution des programmes est effectuée par des scripts dédiés pour plus de simplicité. Ces derniers sont listés ci-dessous :

- **run_physical_stage.sh** démarre la simulation de l'étape physique en exécutant le programme `phys_geo`.
- **extractChemInput.sh** permet de procéder à l'extraction des données d'entrées nécessaire à la simulation de l'étape chimique. Cette extraction est effectuée par le programme `extractChemInput`.
- **run_chemical_stage.sh** démarre et contrôle la simulation de l'étape chimique en exécutant les nombreuses simulations intermédiaires requises. Chaque simulation intermédiaire correspondant à une exécution du programme `chem_geo` associé à un fichier d'entrée contenant les données relatives à un couple event/voxel (cf. section 3.2).
- **run_analyse.sh** procède successivement à l'exécution des programmes `extractSSB`, `DBScan` et `postProcessDBScan` pour analyser les résultats de la simulation des étapes physiques, physico-chimique et chimiques.
- **run.sh** exécute séquentiellement les scripts `run_physical_stage.sh`, `extractChemInput.sh` et `run_chemical_stage.sh`.
- **run_stat_control.sh** prend en charge le lancement de `statControl`.
- **runs.sh** exécute séquentiellement et en boucle les scripts `run.sh` et `run_stat_control.sh`. Le script `runs.sh` contient une incertitude statistique qui est l'incertitude maximale tolérée pour le résultat attendu (2% par défaut).

De manière similaire, la compilation des programmes est prise en charge par les scripts suivants :

- **build.sh** compile `phys_geo`, `extractChemInput`, `chem_geo` ainsi que `extractSSB`, `DBScan` et `PostProcessDBScan`.
- **build_stat_control.sh** compile `statControl`.
- **builds.sh** exécute séquentiellement `build_stat_control.sh` et `build.sh`.

L'utilisation de l'ensemble de ces scripts est facilitée par l'architecture interne de FullSim qui place les scripts et leurs programmes associés dans des modules distincts (dossiers) : `phys_geo`, `extractChemInput`, `chem_geo`, `analysis`, `simu`, `batch`, `statistics`. Ces modules sont imbriqués entre eux et organisés hiérarchiquement selon le modèle décrit par la figure 3.8. Remarquons que cette hiérarchie correspond également à la disposition des dossiers et fichiers de FullSim. FullSim contient trois modules principaux respectivement nommés Simu, Statistics et Batch. Simu est le module en charge de la simulation des étapes physique, physico-chimique et chimique tandis que le module Statistics vérifie que l'incertitude statistique est en deçà d'un seuil défini par l'utilisateur. Lorsque cette incertitude est trop importante, le module Simu est exécuté plusieurs fois de suite à l'intérieur de batchs (module Batch) et les résultats associés sont cumulés de manière à réduire l'incertitude. Le module Simu contient les sous-modules `phys_geo`, `extractChemInput`, `chem_geo`, `analyse` et `input_geometry`. Comme leurs noms l'indiquent, les modules `phys_geo`, `extractChemInput` et `chem_geo` contiennent les programmes et scripts

du même nom. Le module `input_geometry` est constitué de six fichiers générés par DnaFabric et décrivant les cinq voxels ainsi que leurs positionnements dans un noyau complet ("Nucleus.fab2g4dna"). La procédure à suivre pour activer la chaîne de calcul FullSim est

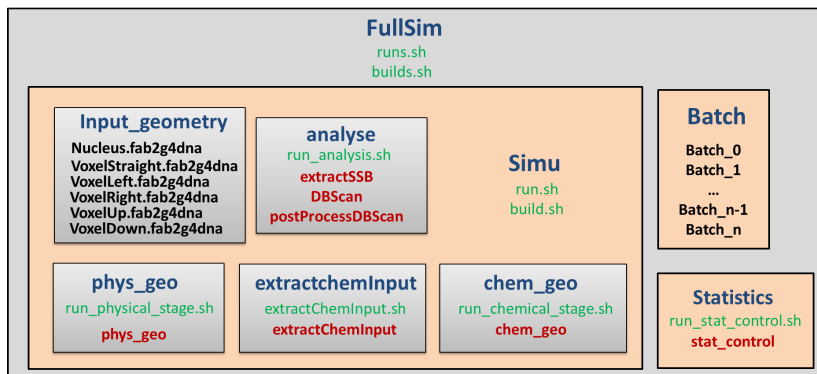


FIGURE 3.8 – Les différents modules contenus dans la chaîne de calcul FullSim. Les modules sont représentés par des carrés et leurs noms sont écrits en bleu. Les noms des programmes sont en rouge tandis que ceux des scripts sont en vert.

détaillée ci-après.

3.3.2 Utilisation de "FullSim"

Le lancement d'une simulation avec la chaîne de calcul FullSim requiert une installation fonctionnelle de Geant4.LDRI.01 ainsi que la présence de fichiers décrivant le noyau cellulaire dans lequel la simulation se déroulera. Une fois ces éléments disponibles, la procédure à suivre pour utiliser FullSim est la suivante :

1. Copier le dossier de FullSim (généralement nommé "FullSim") dans un espace de travail dédié à la simulation à effectuer.
2. Éditer le script `runs.sh`, situé dans le dossier "FullSim" nouvellement créé, pour y spécifier l'incertitude relative tolérée. La simulation se déroulera jusqu'à ce que l'incertitude calculée soit inférieure à celle spécifiée.
3. Éditer le script `run.sh`, situé dans `FullSim/simu`, pour y indiquer le type de particule à lancer, son énergie, l'emplacement de l'installation de Geant4.LDRI.01 ainsi que le nombre d'unités de calculs (threads) à utiliser (MT pour la partie physique).
4. Exécuter le script `builds.sh` pour compiler l'ensemble des programmes de FullSim.
5. Exécuter le script `runs.sh` pour démarrer la simulation.

Dans sa configuration par défaut, FullSim calcule un nombre de DSB moyen par particule primaire dans l'ensemble du noyau cellulaire. C'est également par rapport à ce résultat que l'incertitude statistique est estimée. Cependant, au delà de ce résultat principal, de nombreuses autres informations sont accessibles à la fin de la simulation. Leur accès est plus ou moins aisé selon la situation, voici quelques exemples :

- La totalité des dommages à l'ADN occasionnés lors des étapes physique et chimique sont listés dans des fichiers de sortie dédiés. Parmi ces dommages, se trouve ceux concernant les bases de l'ADN par exemple. Dans un cas où l'obtention d'une incertitude acceptable n'a requis que le lancement d'un seul batch alors ces fichiers sont directement accessibles dans `FullSim/batch/batch_0/analyse/extractSSB/extractSSB-build/output`.

- Le fichier de sortie directement généré par le programme DBScan contient de nombreuses informations liées au degré de complexité des différents dommages à l'ADN analysés. Seuls les SSBs et DSBs en sont extraits par défaut mais une analyse plus poussée permettrait d'obtenir des informations sur le degré de complexité de chaque SSB ou DSB. C'est à dire, par exemple, le nombre d'ionisations, excitations ou réactions constitutives d'un SSB ou DSB donné. Le fichier de sortie de DBScan se trouve dans *FullSim/batch/batch_0/analyse/DBScan/output*.
- Les informations précédemment citées et concernant les dommages à l'ADN sont accessibles chromosome par chromosome.

Enfin, il est possible d'exécuter une analyse entièrement nouvelle à partir des résultats "bruts" générés par *phys_geo* et *chem_geo* en écrivant un nouveau programme dédié ou en modifiant un programme déjà existant. Cette modification doit être effectuée avant le point 4 de la procédure précédemment décrite. Une telle modification permet, par exemple, d'obtenir un nombre de fragments d'ADN en tant que résultat au lieu d'un nombre de DSBs. Dans la plupart des cas, ces éventuelles modifications ne concernent que l'étape d'analyse et l'architecture modulaire de FullSim rend alors possible de n'exécuter à nouveau que cette dernière étape d'analyse. Ainsi, une seule simulation des étapes physique, physico-chimique et chimique peut permettre d'obtenir de nombreux résultats au moyen de différentes analyses successives. La durée requise pour effectuer une analyse étant de quelques minutes et celle de la simulation des trois étapes de quelques jours, l'exécution de l'étape d'analyse uniquement occasionne un gain de temps très significatif.

La figure 3.9 est un logigramme illustrant l'enchaînement des scripts et programmes exécutés lors d'une simulation effectuée avec la chaîne de calcul FullSim. La compilation de l'ensemble des programmes inclus dans FullSim n'est pas incluse dans ce logigramme, elle doit cependant être effectuée au préalable via le script *builds.sh*. Le début de la simulation correspond à l'exécution du script maître *runs.sh*. Ce script va créer le dossier *batch_0* à l'emplacement suivant : *FullSim/batch/*. L'ensemble du contenu du répertoire *FullSim/simu* est ensuite copié dans *batch_0*, ceci comprend à la fois les sources, les exécutable des différents programmes et les scripts de *simu*. Une fois cette copie effectuée, le *batch_0* est prêt à être lancé. Le lancement d'un batch correspond à l'appel du script *run.sh* contenu dans ce dernier. Ainsi, dans le cas de *batch_0*, le script *batch_0/run.sh* est appelé par le script maître. Le script *run.sh* appelé va alors démarrer successivement la simulation des étapes physique, physico-chimique et chimique ainsi que l'analyse des résultats et ceci pour un tirage de 1000 particules primaires dans le noyau cellulaire. Dans un premier temps, le script *run_physical_stage.sh* est appelé et exécute le programme *phys_geo* qui va simuler le transport physique des 1000 particules introduites dans la simulation. Cette simulation est effectuée en parallèle sur plusieurs unités de calcul (Geant4 mode MT). Dans un second temps, le script *extract.sh* est appelé et va exécuter le programme *extractChemInput* pour préparer les données d'entrée nécessaires à la simulation des étapes physico-chimique et chimique, soit les fichiers représentatifs des couples *event/voxel*. Enfin et dans un troisième temps, le script *run_chemical_stage.sh* est appelé avec, en tant que données d'entrées, tous les couples *event/voxel* précédemment générés. Ce script se charge d'attribuer à chaque unité de calcul (thread) disponible une instance du programme *chem_geo* qui correspond à un couple *event/voxel*. Le script balaye à chaque seconde l'ensemble des unités de calcul pour détecter leur éventuelle libération lorsque le programme *chem_geo* qui leur est attribué se termine. Si une unité de calcul est libérée alors un nouveau programme *chem_geo* lui est attribué et ce jusqu'à épuiser tous les fichiers d'entrées (couple *event/voxel*) disponibles. Enfin, le script *run_analyse.sh* est appelé pour déclencher l'analyse des résultats calculés

dans le batch. Ce script va exécuter successivement les programmes `extractSSB`, `DBScan` et `postProcessDBScan`. A ce stage, l'exécution du `batch_0` est terminée et la pertinence statistique des résultats contrôlée au moyen d'un appel au script `run_stat_control.sh` qui déclenche l'exécution du programme `statControl`. Ce programme calcule la distribution du nombre de DSBs obtenus par particule primaire à partir des données générées dans le batch récemment exécuté. Si des données concernant de précédents batch sont présentes, alors `statControl` les cumule au sein de la distribution qu'il génère. Un nombre de DSBs moyen par particule primaire envoyé est estimé à partir de cette distribution et une incertitude statistique lui est associée. Cette incertitude est comparée à celle spécifiée par l'utilisateur préalablement au lancement de `FullSim` et, si elle est supérieure, un nouveau batch est créé. Dans le cas où l'incertitude obtenue est inférieure à celle spécifiée alors la chaîne de calcul `FullSim` s'arrête et le calcul se termine.

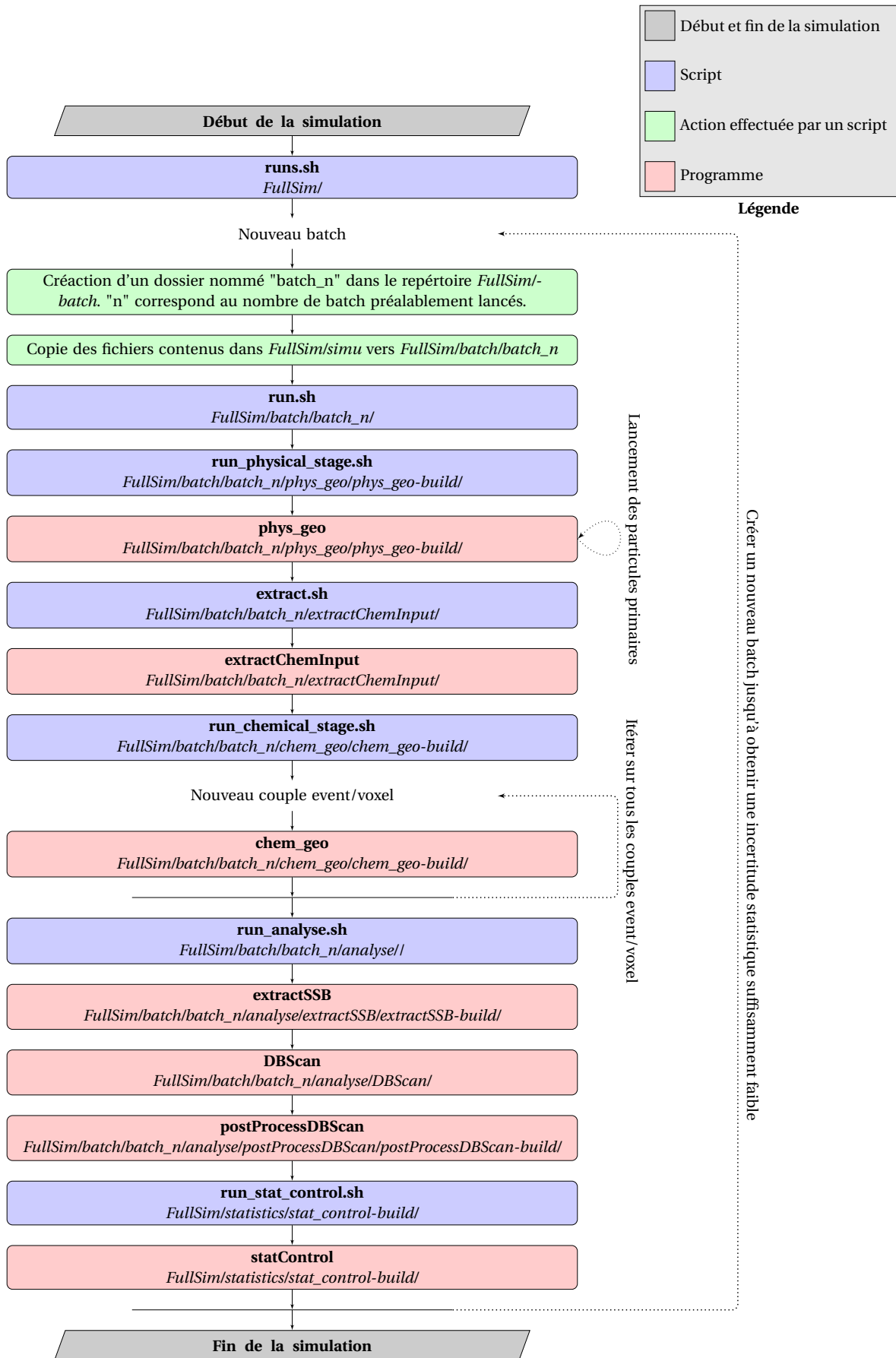


FIGURE 3.9 – Logigramme du fonctionnement de la chaîne de calcul "FullSim".

Références

- ABOLFATH, R. M., D. J. CARLSON, Z. J. CHEN et R. NATH. 2013, «A molecular dynamics simulation of DNA damage induction by ionizing radiation», *Physics in Medicine and Biology*, vol. 58, n° 20, p. 7143–7157. [114](#), [121](#)
- ALEXANDER, F., C. VILLAGRASA, H. RABUS et J. J. WILKENS. 2015, «Local weighting of nanometric track structure properties in macroscopic voxel geometries for particle beam treatment planning», *Physics in Medicine and Biology*, vol. 60, n° 23, p. 9145–9156. [106](#)
- AYDOGAN, B., D. T. MARSHALL, S. G. SWARTS, J. E. TURNER, A. J. BOONE, N. G. RICHARDS et W. E. BOLCH. 2002, «Site-specific OH attack to the sugar moiety of DNA : a comparison of experimental data and computational simulation», *Radiation research*, vol. 157, n° 1, p. 38–44. [120](#)
- BAEK, W. Y., A. ARNDT, M. U. BUG, H. RABUS et M. WANG. 2013, «Total electron-scattering cross sections of pyrimidine», *Physical Review A*, vol. 88, n° 3. [100](#)
- BAEK, W. Y., M. BUG, H. RABUS, E. GARGIONI et B. GROSSWENDT. 2012, «Differential elastic and total electron scattering cross sections of tetrahydrofuran», *Physical Review A*, vol. 86, n° 3. [100](#)
- BALASUBRAMANIAN, B., W. K. POGOZELSKI et T. D. TULLIUS. 1998, «DNA strand breaking by the hydroxyl radical is governed by the accessible surface areas of the hydrogen atoms of the DNA backbone», *Proceedings of the National Academy of Sciences*, vol. 95, n° 17, p. 9738–9743. [120](#), [121](#)
- BALD, I., E. ILLENBERGER et J. KOPYRA. 2012, «Damage of DNA by Low Energy Electrons», *Journal of Physics : Conference Series*, vol. 373, p. 012 008. [115](#)
- BANTSAR, A. 2012, «Ionization cluster size distributions created by low energy electrons and alpha particles in nanometric track segment in gases», *arXiv preprint arXiv :1207.6893*. [106](#)
- BERNAL, M., D. SIKANSI, F. CAVALCANTE, S. INCERTI, C. CHAMPION, V. IVANCHENKO et Z. FRANCIS. 2013, «An atomistic geometrical model of the B-DNA configuration for DNA–radiation interaction simulations», *Computer Physics Communications*, vol. 184, n° 12, p. 2840–2847. [110](#)
- BOUDAIFFA, B., P. CLOUTIER, D. HUNTING, M. A. HUELS et L. SANCHE. 2000, «Resonant formation of DNA strand breaks by low-energy (3 to 20 eV) electrons», *Science*, vol. 287, n° 5458, p. 1658–1660. [115](#)
- BRUN, R. et F. RADEMAKERS. 1997, «ROOT—an object oriented data analysis framework», *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 389, n° 1, p. 81–86. [113](#)
- BUG, M. U. 2014, *Nanodosimetric particle track simulations in water and DNA media*, thèse de doctorat, University of Wollongong, School of Physics. [100](#)
- BUG, M. U., H. RABUS et A. B. ROSENFELD. 2012, «Electron emission from amorphous solid water after proton impact : Benchmarking PTra and Geant4 track structure Monte Carlo simulations», *Radiation Physics and Chemistry*. [106](#)

- BUXTON, G. V., C. L. GREENSTOCK, W. P. HELMAN et A. B. ROSS. 1988, «Critical Review of rate constants for reactions of hydrated electrons, hydrogen atoms and hydroxyl radical in Aqueous Solution», *J. Phys. Chem*, vol. 17, n° 513. [119](#)
- CELIBERTO, R., R. JANEV, J. WADEHRA et J. TENNYSON. 2012, «Dissociative electron attachment to vibrationally excited H₂ molecules involving the resonant Rydberg electronic state», *Chemical Physics*, vol. 398, p. 206–213. [115](#)
- DOS SANTOS, M. 2013, *Modélisation de la topologie des dépôts d'énergie créés par un rayonnement ionisant à l'échelle nanométrique dans les noyaux cellulaires et relation avec les événements précoces radio-induits*, thèse de doctorat, Bordeaux 1, Ecole doctorale des sciences physiques et de l'ingénieur. [100](#)
- FRIEDLAND, W., P. BERNHARDT, P. JACOB et H. G. PARETZKE. 2002, «Simulation of DNA damage after proton and low LET irradiation», *Radiation protection dosimetry*, vol. 99, n° 1-4, p. 99–102. [121](#)
- FRIEDLAND, W., P. JACOB, P. BERNHARDT, H. G. PARETZKE et M. DINGFELDER. 2003, «Simulation of DNA damage after proton irradiation», *Radiation Research*, vol. 159, p. 401–410. [115](#)
- FRIEDLAND, W., H. G. PARETZKE, F. BALLARINI, A. OTTOLENGHI, G. KRETH et C. CREMER. 2008, «First steps towards systems radiation biology studies concerned with DNA and chromosome structure within living cells», *Radiation and Environmental Biophysics*, vol. 47, n° 1, p. 49–61. [110](#)
- GARTY, G., R. SCHULTE, S. SHCHEMELININ, C. LELOUP, G. ASSAF, A. BRESKIN, R. CHECHIK, V. BASHKIROV, J. MILLIGAN et B. GROSSWENDT. 2010, «A nanodosimetric model of radiation-induced clustered DNA damage yields», *Physics in medicine and biology*, vol. 55, n° 3, p. 761. [106](#)
- GOODHEAD, D. 1994, «Initial events in the cellular effects of ionizing radiations : clustered damage in DNA», *Int. J. Radiat. Biol.*, vol. 65, n° 1, p. 7–17. [115](#)
- GROSSWENDT, B. 2002, «Formation of ionization clusters in nanometric structures of propane-based tissue-equivalent gas or liquid water by electrons and alpha-particles», *Radiation and Environmental Biophysics*, vol. 41, n° 2, p. 103–112. [100](#)
- KREIPL, M. S., W. FRIEDLAND et H. G. PARETZKE. 2009, «Interaction of ion tracks in spatial and temporal proximity», *Radiation and Environmental Biophysics*, vol. 48, n° 4, p. 349–359. [119](#)
- LAPORTA, V., R. CELIBERTO et J. TENNYSON. 2015, «Dissociative electron attachment and electron-impact resonant dissociation of vibrationally excited O₂ molecules», *Physical Review A*, vol. 91, n° 1. [115](#)
- LINERT, I., M. DAMPC, B. MIELEWSKA et M. ZUBEK. 2012, «Cross sections for ionization and ionic fragmentation of pyrimidine molecules by electron collisions», *The European Physical Journal D*, vol. 66, n° 1. [100](#)
- MILLIGAN, J. R., C. C. L. WU, J. A. AGUILERA et J. F. WARD. 1996, «Characterization of the reaction rate coefficient of DNA with the hydroxyl radical», *Radiation research*, vol. 146, n° 5, p. 510–513. [119](#)

- NIKJOO, H. et P. GIRARD. 2012, «A model of the cell nucleus for DNA damage calculations», *International Journal of Radiation Biology*, vol. 88, n° 1-2, p. 87–97. [110](#)
- NIKJOO, H., P. O'NEILL, D. GOODHEAD et M. TERRISOL. 1997, «Computational modelling of low-energy electron-induced DNA damage by early physical and chemical events», *International Journal of Radiation Biology*, vol. 71, n° 5, p. 467–483. [115](#)
- NIKJOO, H., P. O'NEILL, W. E. WILSON et D. T. GOODHEAD. 2001, «Computational approach for determining the spectrum of DNA damage induced by ionizing radiation», . [115](#)
- PARBERRY, I. 2014, «Designer worlds : Procedural generation of infinite terrain from real-world elevation data», *Journal of Computer Graphics Techniques*, vol. 3, n° 1. [111](#)
- REZAEI, M., L. SANCHE et D. J. HUNTING. 2013, «Cisplatin Enhances the Formation of DNA Single- and Double-Strand Breaks by Hydrated Electrons and Hydroxyl Radicals», *Radiation Research*, vol. 179, n° 3, p. 323–331. [114](#)
- ROOTS, R. et S. OKADA. 1975, «Estimation of Life Times and Diffusion Distances of Radicals Involved in X-Ray-Induced DNA Strand Breaks or Killing of Mammalian Cells», *Radiation Research*, vol. 64, p. 306–320. [119](#)
- SHAFRANYOSH, I. I., M. I. SUKHOVIYA et M. I. SHAFRANYOSH. 2006, «Absolute cross sections of positive- and negative-ion production in electron collision with cytosine molecules», *Journal of Physics B : Atomic, Molecular and Optical Physics*, vol. 39, n° 20, p. 4155–4162. [100](#)
- SWARTS, S. G., D. C. GILBERT, K. K. SHARMA, Y. RAZSKAZOVSKIY, S. PURKAYASTHA, K. A. NAUMENKO et W. A. BERNHARD. 2007, «Mechanisms of Direct Radiation Damage in DNA, Based on a Study of the Yields of Base Damage, Deoxyribose Damage, and Trapped Radicals in d(GCACGCGTGC)», *Radiation Research*, vol. 168, n° 3, p. 367–381. [115](#)
- TERRISOL, M. et J. M. VRIGNEAUD. 2001, «Analogue Monte Carlo to model Radiation-induced DNA damage», dans *Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications*, Springer, p. 261–265. [110](#)
- VILLAGRASA, C., Z. FRANCIS et S. INCERTI. 2010, «Physical models implemented in the GEANT4-DNA extension of the GEANT-4 toolkit for calculating initial radiation damage at the molecular level», *Radiation Protection Dosimetry*, vol. 143, n° 2-4, p. 214–218. [106](#)

Chapitre 4

Dommmages à l'ADN obtenus par la simulation de l'irradiation d'un noyau cellulaire

« La phrase la plus excitante à entendre en science, celle qui annonce de nouvelles découvertes, n'est pas "Eurêka" (j'ai trouvé!), mais plutôt "Tiens, c'est marrant..." »

Isaac Asimov

Sommaire

4.1 Irradiation d'un noyau de cellule fibroblaste : comparaison avec des données de la littérature	133
4.1.1 Configuration des simulations	134
4.1.2 Résultats : comparaison des résultats de la simulation avec les données publiées sur les fragments d'ADN	137
4.1.3 Discussion	143
4.2 Influence des sections efficaces de l'ADN	146
4.2.1 Configuration des simulations	146
4.2.2 Résultats : influence du matériau composite ADN sur le nombre d'ionisations à l'échelle de 10 paires de nucléotides	148
4.2.3 Résultats : influence du matériau composite ADN à l'échelle du noyau de cellule fibroblaste	149
4.2.4 Discussion	152
4.3 Sensibilité de la simulation	155
4.3.1 Configuration des simulations	155
4.3.2 Résultats : sélection des dommages directs	158
4.3.3 Résultats : durée simulée de l'étape chimique	159
4.3.4 Résultats : influence des voxels isolés	160
4.3.5 Discussion	162
Références	168

Ce chapitre présente les premiers résultats obtenus avec la chaîne de calcul "FullSim" introduite dans le chapitre précédent (cf. chapitre 3). La majorité des simulations réalisées à cet effet reproduisent l'irradiation d'un noyau de cellule fibroblaste avec des protons mono-énergétiques. Cette irradiation a été simulée à plusieurs reprises dans le cadre de trois objectifs distincts : effectuer une comparaison avec des résultats rapportés dans la littérature, estimer l'influence des nouvelles sections efficaces de l'ADN et quantifier la sensibilité des résultats simulés à certains paramètres du calcul. Ces trois objectifs constituent les trois sections de ce chapitre et représentent, à notre connaissance, la première utilisation de Geant4-DNA pour la simulation des étapes physique, physico-chimique et chimique dans un noyau cellulaire contenant un génome humain complet.

4.1 Irradiation d'un noyau de cellule fibroblaste : comparaison avec des données de la littérature

Comme indiqué au chapitre précédent, les paramètres de la chaîne de calcul FullSim sont liés aux hypothèses et limitations incluses dans la modélisation et, à ce titre, les valeurs attribuées par défaut aux paramètres de notre simulation doivent être testées. Les simulations et résultats présentés dans cette section ont pour finalité la réalisation de ces tests. Ces derniers consistent à comparer les résultats issus de notre chaîne de calcul avec des données publiées dans la littérature. Dans ce contexte, les données expérimentales constituent une référence idéale mais rare car elles sont issues d'expérimentations en radiobiologie très spécifiques et peu nombreuses. De plus, les données ainsi obtenues sont étroitement associées à des conditions expérimentales précises ce qui implique qu'elles doivent être analysées en fonction de ces dernières : cible biologique spécifique, biais introduits par le dispositif expérimental, normalisation des données etc. Ainsi, par exemple, les résultats obtenus par immunofluorescence (γ H2AX, cf. section 1.4.3) et ceux obtenus par électrophorèse en champ pulsé [FRANKENBERG et al., 1999] ne peuvent être analysés de manière identique. Finalement, ces spécificités complexifient la comparaison entre résultats simulés et résultats expérimentaux. Pour limiter les difficultés associées à cette analyse, le choix a été fait dans ce travail de ne considérer que les données expérimentales obtenues dans un cadre précis : l'analyse par électrophorèse en champ pulsé des fragments d'ADN générés par des irradiations de cellules avec des protons [BELLI et al., 2001, 2000; CAMPA et al., 2005; FRANKENBERG et al., 1999]. Ce cadre a été choisi car il est aisément reproductible dans nos simulations et, de plus, plusieurs autres simulations ont déjà été ainsi réalisées : [FRIEDLAND et al., 2003] et [NIKJOO et al., 2001]. Les résultats générés par ces simulations sont également pris en compte dans cette section et comparés avec ceux obtenus avec notre chaîne de calcul.

Remarquons qu'il existe d'autres simulations basées sur Geant4 et visant à simuler l'apparition des dommages précoces à l'ADN suite à une irradiation de cellule. Chacune de ces simulations possède ses propres hypothèses correspondant à un ensemble de paramètres dont les valeurs sont spécifiques et ne prennent généralement pas en compte la simulation des étapes physico-chimique et chimique [BERNAL et al., 2015; SANTOS et al., 2014]. A ce propos, la simulation développée dans ce travail a précisément pour particularité de combiner la prise en compte des étapes physique, physico-chimique et chimique avec une modélisation géométrique du génome humain détaillée jusqu'à l'échelle moléculaire.

4.1.1 Configuration des simulations

Géométrie

Les simulations présentées modélisent l'irradiation d'un noyau de cellule fibroblaste généré par DnaFabric et introduit en section 2.3.3. Ce noyau contient un génome humain mâle relaxé, soit 46 chromosomes (23 paires) totalisant un peu plus de 6 milliards de paires de nucléotides. L'enveloppe extérieure du noyau est une ellipsoïde dont les dimensions sont données dans le tableau 2.7 (cf. section 2.3.3) et rappelées en figure 4.1. Un fichier (.fab2g4dna) descriptif de ce noyau ainsi que les fichiers associés aux voxels qu'il contient sont générés par DnaFabric puis placés à l'intérieur d'un dossier nommé *simu/input_geometry* faisant partie de la chaîne de calcul FullSim. Toutefois, pour s'approcher des simulations décrites dans [FRIEDLAND et al., 2003] et [NIKJOO et al., 2001], le module *phys_geo* est modifié pour ne prendre en compte qu'un seul matériau lors du transport de particules : l'eau liquide. L'ADN sera cependant pris en compte en tant que matériau spécifique dans la section 4.2.

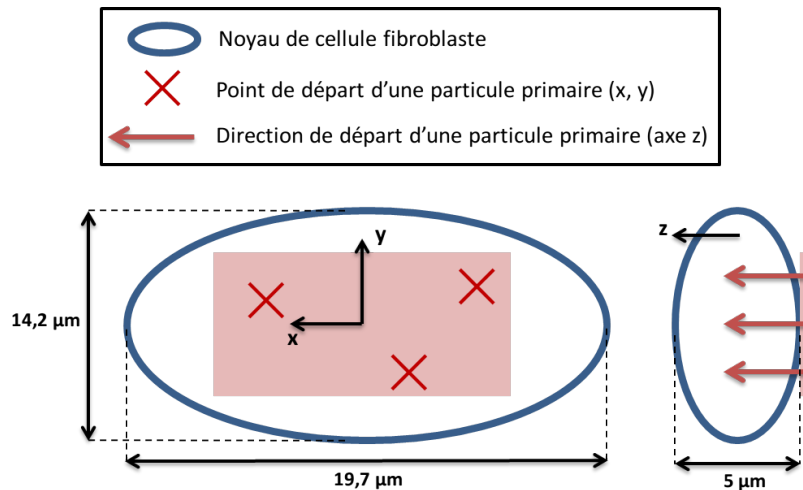


FIGURE 4.1 – Dimensions du noyau de cellule fibroblaste et configuration de la source de particules utilisée. Cette source est placée sur la surface représentée par le rectangle rouge et les particules primaires sont tirées dans la direction indiquée par les flèches rouges.

Particules primaires

Les particules primaires utilisées dans ces simulations sont des protons mono-énergétiques de différents TEL (keV/µm) : 2,8 ; 5 ; 8,7 ; 10,3 ; 12,8 ; 17,4 ; 21,3 ; 28,9 ; 33,9 ; 37,6 et 42,6. Les énergies cinétiques des protons associés à ces TEL sont présentées dans le tableau 4.1. Notons que les protons traversent le noyau de fibroblaste selon l'axe Z de l'ellipsoïde comme illustré en figure 4.1. et qu'ils sont "tirés" depuis une surface rectangulaire de 16 et 12 µm de côtés afin d'éviter les bords du noyau cellulaire. Cette surface est positionnée sur l'axe Z à -2,5 µm, soit juste en "dessous" du noyau cellulaire. A partir de cette surface, les particules primaires sont dirigées selon l'axe Z, c'est à dire en direction du noyau cellulaire. La configuration de cette source est illustrée en figure 4.1.

TABLEAU 4.1 – Énergies cinétiques et TEL des protons utilisés en tant que particules primaires dans la simulation d'un noyau de cellule fibroblaste. Les TEL présentés ont été obtenus à partir de simulations Geant4 menées dans le cadre de ce travail.

Énergie (MeV)	TEL (keV/ μm)
0,5	42,6
0,7	37,6
0,8	33,9
1	28,9
1.5	21,3
2	17,4
3	12,8
4	10,3
5	8,7
10	5
20	2,8

Paramètres des simulations

Les valeurs des paramètres associés aux simulations exécutées sont celles attribuées par défaut et mentionnées au chapitre 3. Pour rappel, elles sont résumées ci-dessous :

- Toute ionisation ou excitation (électronique ou vibrationnelle) ayant lieu dans une molécule de phosphate, de désoxyribose ou dans leurs couches d'hydratation respectives sera prise en compte pour le calcul des dommages directs.
- Condition sur l'apparition d'un SSB_{direct} : énergie déposée de plus de 17,5 eV dans le sucre et le phosphate d'un nucléotide (squelette de l'ADN).
- Condition sur l'apparition d'un SSB indirect : toute réaction entre un radical OH[•] et une molécule de désoxyribose a 42% de chances d'entraîner un SSB indirect.
- Une cassure double brin (DSB) est formée de, au minimum, deux SSB séparés entre eux par moins de 10 paires de nucléotides et situés sur des brins opposés.
- Le temps simulé lors de l'étape chimique est de 2,5 ns.
- Les molécules de phosphate de sont pas prises en compte lors de la simulation chimique.
- Durant l'étape chimique, le complexe de protéine histone est représenté par une sphère de 2,4 nm de rayon qui absorbe tout ce qui entre en contact avec elle.

Notons que la condition sur l'apparition d'un SSB_{direct} peut être modifiée en suivant l'exemple de celle utilisée dans le code PARTRAC [FRIEDLAND et al., 2003]. Cette condition devient alors une probabilité d'acceptation variant linéairement entre 5 à 37,5 eV et caractérisée par : $P_{accept}(E_{dep} \leq 5 \text{ eV}) = 0$ et $P_{accept}(E_{dep} \geq 37,5 \text{ eV}) = 1$.

Traitement des résultats produits

Les données expérimentales utilisées dans cette section proviennent d'expériences consistant à irradier des noyaux cellulaires pour mesurer les fragments d'ADN générés. Il est possible d'accéder au nombre de DSB induits par l'irradiation à partir des données relatives à ces fragments en considérant qu'un fragment est produit à partir de deux DSB localisés sur un même filament d'ADN. Cependant, les dispositifs expérimentaux utilisés

[BELLI et al., 2001, 2000; CAMPA et al., 2005; FRANKENBERG et al., 1999] ne peuvent détecter les fragments de petites taille en raison de leurs résolutions limitées. Finalement, les données expérimentales publiées sont des nombres moyens de DSB par Gy et par Gbp ($1 \cdot 10^9$ paires de nucléotides) obtenus à partir des fragments d'ADN observés lors d'irradiation où de fortes doses sont délivrées dans le noyau cellulaire (> 100 Gy). Pour chaque type de particule primaire, plusieurs irradiations sont effectuées de manière à délivrer plusieurs doses et pouvoir ensuite calculer le nombre de DSB/Gy/Gbp au moyen d'une régression linéaire illustrée en figure 4.2. A ce propos, notons que l'incertitude associée aux résultats expérimentaux ainsi obtenus est élevée.

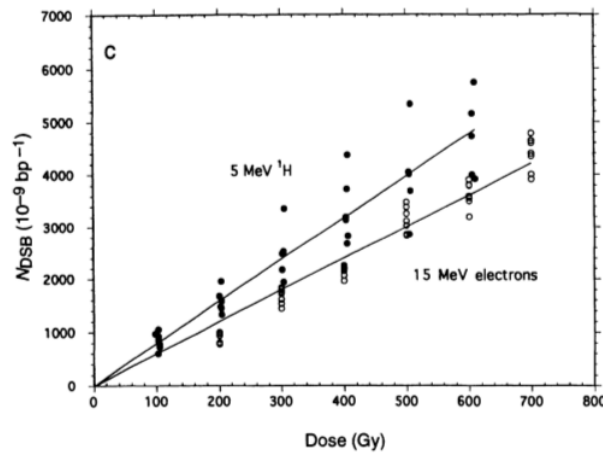


FIGURE 4.2 – Régression linéaire effectuée à partir de plusieurs irradiations effectuées avec des protons de 5 MeV et des électrons de 15 MeV [FRANKENBERG et al., 1999].

Pour permettre à FullSim de générer des résultats équivalents, une étape supplémentaire a été intégrée par rapport à celles décrites en section 3.3.2 : le calcul des tailles de fragments qui est effectué à partir des DSB simulés et de leurs positions dans le génome. Plus précisément, chaque DSB calculé peut être localisé dans le génome humain au moyen de deux identifiants (attribués à chaque DSB) : numéro du chromosome sur lequel se situe le DSB et numéro de la paire de nucléotides (à l'échelle du chromosome) située au "milieu" du DSB. A partir des DSB ainsi localisés, une liste de fragments de diverses tailles est constituée et les fragments dont la taille est inférieure à un seuil s_{bp} sont considérés comme "indétectables". Le nombre de fragments indétectables obtenus est ensuite soustrait au nombre de DSB calculés pour obtenir un nombre de DSB par événement (i.e. par particule primaire). Ainsi, nous sommes en mesure de déterminer les fragments qui pourraient être compatibles avec un dispositif expérimental donné. L'application de l'équation 4.1 permet ensuite de passer des [DSB/event] aux [DSB/Gy/Gbp].

$$N_{\text{DSB/Gy/Gbp}}(s_{bp}) = N_{\text{DSB/event}}(s_{bp}) \cdot \frac{E_{1\text{Gy}}}{\bar{l} \cdot \text{TEL}(E_p) \cdot n} \quad (4.1)$$

avec $N_{\text{DSB/Gy/Gbp}}(s_{bp})$ le nombre de fragments par Gy et Gbp obtenus pour un seuil s_{bp} , $N_{\text{DSB/event}}(s_{bp})$ le nombre de fragments par particule primaire (event), $E_{1\text{Gy}}$ l'énergie (eV) à déposer dans le noyau pour atteindre 1 Gy (J/kg), \bar{l} la distance moyenne parcourue par une particule primaire dans le noyau, $\text{TEL}(E_p)$ le TEL de la particule primaire d'énergie E_p et, enfin, n le nombre de Gbp contenu dans le noyau. Dans notre cas, des simulations

spécifiques ont été menées pour déterminer ces paramètres et nous avons obtenu :

$$\begin{cases} E_{1Gy} & = 4570 \text{ eV} \\ \bar{l} & = 3,5 \text{ } \mu\text{m} \\ n & = 6 \text{ Gbp} \end{cases}$$

Chaque transfert d'énergie linéique (TEL) correspondant à une énergie des protons primaires a été calculé avec Geant4-DNA et est présenté dans le tableau 4.1.

Outre le nombre de DSB générés, les simulations réalisées dans ce travail donnent également accès aux SSB (SSB_{tot}) qui les constituent. En effet, ces derniers sont stockés à part lors de l'exécution du module d'analyse présent dans FullSim (cf. l'étape "extractSSB" de la figure 3.9, section 3.3.2). A ce niveau, il est possible de distinguer les SSB générés lors de la simulation de l'étape physique (SSB_{direct}) de ceux produits lors de la simulation de l'étape chimique ($SSB_{indirect}$). Les quantités de SSB_{tot} , SSB_{direct} et $SSB_{indirect}$ obtenus dans ce travail sont également présentées et ces résultats sont organisés en fonction des TEL des protons primaires de manière à être comparés aux données du même type disponibles dans [FRIEDLAND et al., 2003].

Enfin, les informations détaillées concernant les SSB ont pu être utilisées pour déterminer la complexité des DSB simulés avec notre simulation. Cette complexité correspond à la quantité de SSB (direct ou indirect) qui constitue un DSB. Notons qu'elle ne peut être inférieure à 2 car il s'agit du nombre minimal de SSB requis pour former un DSB.

4.1.2 Résultats : comparaison des résultats de la simulation avec les données publiées sur les fragments d'ADN

La figure 4.3 présente les nombres moyens de DSB/Gy/Gbp calculés dans ce travail avec les valeurs attribuées par défaut aux paramètres de la simulation (cf. section 4.1.1). Les nombres moyens de DSB/Gy/Gbp détectés expérimentalement par [FRANKENBERG et al., 1999], [BELLI et al., 2001, 2000] et [CAMPÀ et al., 2005] sont également illustrés sur cette figure. Notons que ce sont les conditions expérimentales de [FRANKENBERG et al., 1999] qui sont reproduites dans nos simulations, soit un noyau de cellule fibroblaste et une taille minimale de fragment de 10000 bp. En effet, les données de [BELLI et al., 2000], [BELLI et al., 2001] et [CAMPÀ et al., 2005] sont obtenues avec des cellules V-79 (hamster chinois) et la taille minimale des fragments détectés est de 23000 bp dans [BELLI et al., 2001, 2000]. Bien qu'obtenues dans des conditions différentes, ces données sont tout de même illustrées en figure 4.3 pour rendre compte de la dispersion de résultats expérimentaux. Les données expérimentales indiquent que le nombre moyen de DSB/Gy/Gbp augmente légèrement avec le TEL : les données de [FRANKENBERG et al., 1999] débutent à 8,16 DSB/Gy/Gbp pour un TEL de 7,9 keV/ μ m et atteignent 12,22 DSB/Gy/Gbp avec un TEL de 35 keV/ μ m. Remarquons que les résultats de FullSim augmentent également avec le TEL : de 4,8 DSB/Gy/Gbp pour un TEL de 2,8 keV/ μ m jusqu'à 13 DSB/Gy/Gbp pour un TEL de 42,6 keV/ μ m. Les résultats de [BELLI et al., 2001, 2000; CAMPÀ et al., 2005] se situent autour de ceux de [FRANKENBERG et al., 1999] illustrant ainsi les incertitudes associées à ce type de données. Les résultats simulés dans ce travail et présentés en figure 4.3 sont en bon accord avec les résultats expérimentaux disponibles pour des protons. Cet accord est particulièrement bon avec les résultats de [FRANKENBERG et al., 1999] lorsque des TEL supérieurs à 20 keV/ μ m sont considérés. Il est intéressant de remarquer que pour un TEL de 11 keV/ μ m, les résultats simulés sont inférieurs de 25% à ceux de [FRANKENBERG et al., 1999] mais supérieurs de 20% à ceux de [BELLI et al., 2001, 2000].

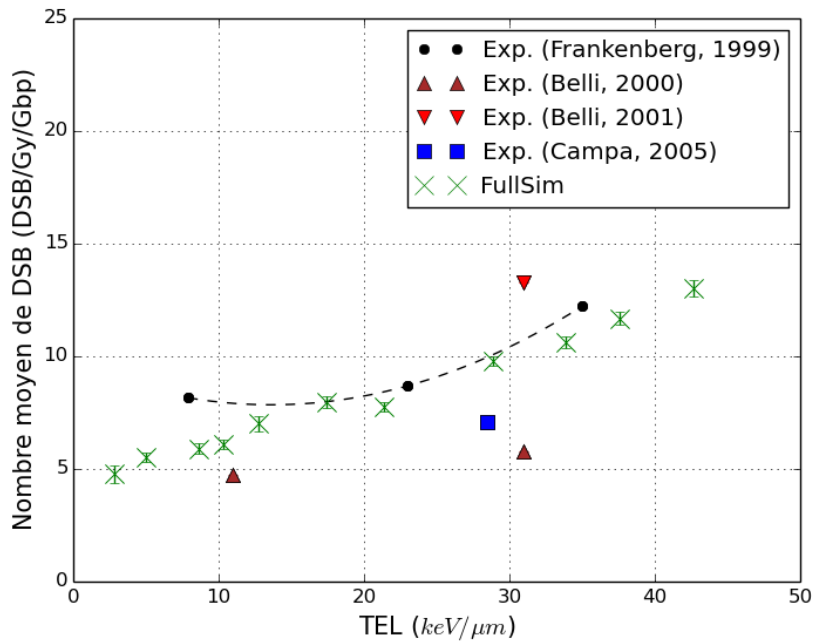


FIGURE 4.3 – Nombres moyens de DSB détectés expérimentalement et simulés dans ce travail lors d'une irradiation avec des protons de différentes énergies. Les données de [FRANKENBERG et al., 1999] sont représentées par des points noirs, des triangles rouges sont utilisés pour les données de [BELLI et al., 2001, 2000] et des carrés bleus pour celles de [CAMPA et al., 2005]. Les résultats obtenus avec FullSim sont présentés par des "x" verts et ont été obtenus avec les valeurs attribuées par défaut aux différents paramètres de notre chaîne de calcul (cf. section 4.1.1).

La figure 4.4 effectue une comparaison entre les résultats simulés décrits dans [NIKJOO et al., 2001] et [FRIEDLAND et al., 2003], respectivement obtenus avec les codes de simulation KURBUC et PARTRAC, et ceux générés par notre chaîne de calcul. Comme précédemment, nos simulations sont utilisées avec leur configuration par défaut. Notons que cette configuration contient des paramètres dont les valeurs sont différentes de celles utilisées dans les codes de simulations KURBUC et PARTRAC. Par exemple et entre autres, il est indiqué dans [FRIEDLAND et al., 2003] que le critère de sélection des SSB_{direct} n'est pas un seuil en énergie déposée de 17,5 eV mais une probabilité d'acceptation variant linéairement entre 5 et 37,5 eV. La figure 4.4 montre que, dans l'ensemble, les nombres moyens de DSB simulés augmentent avec le TEL quelque soit le code de simulation utilisé. Notons cependant que cette augmentation semble linéaire avec KURBUC et notre chaîne de calcul mais pas avec PARTRAC. En effet, l'augmentation du nombre moyen de DSB/Gy/Gbp avec PARTRAC est moins importante pour les TEL supérieurs à 25 keV/μm. Les résultats de [NIKJOO et al., 2001] sont plus élevés que ceux obtenus avec FullSim et PARTRAC. L'écart entre les résultats de [NIKJOO et al., 2001] et ceux de notre simulation oscille constamment entre 5 et 10 DSB/Gy/Gbp pour l'ensemble des TEL considérés. Les résultats simulés dans ce travail sont supérieurs à ceux obtenus avec PARTRAC lorsque le TEL est inférieur à 40 keV/μm et équivalents lorsque le TEL est de 40 keV/μm. Notons que l'écart entre les résultats de PARTRAC et de FullSim est maximal pour un TEL de 2,8 keV/μm. En effet, avec ce TEL, les résultats de PARTRAC sont plus élevés de 71% par rapport à ceux de notre simulation. Cet écart diminue ensuite lorsque le TEL augmente pour atteindre 0 avec un TEL de 40 keV/μm.

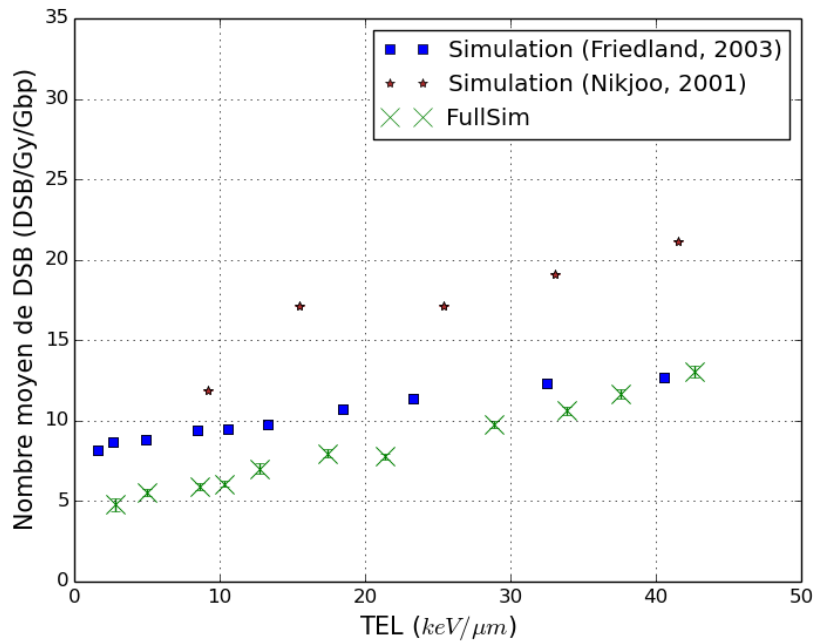
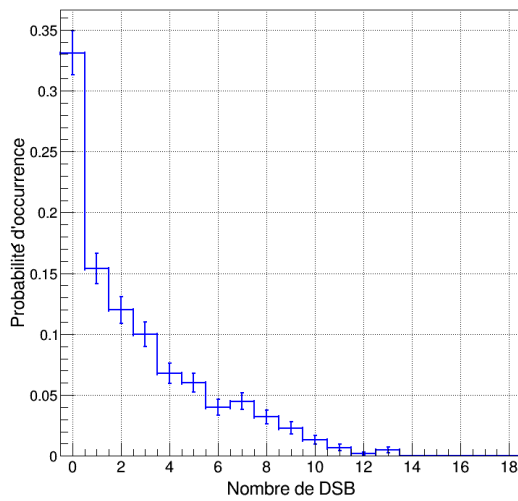


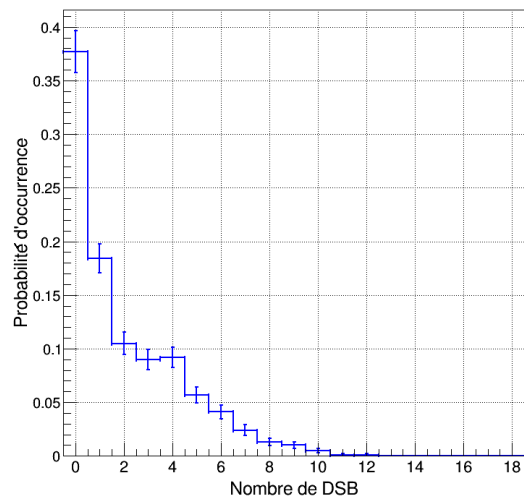
FIGURE 4.4 – Nombre moyen de DSB simulé issu de la littérature et de FullSim. Les résultats de [NIKJOO et al., 2001] sont représentés par des étoiles marrons tandis que des carrés bleus sont utilisés pour ceux de [FRIEDLAND et al., 2003]. Les "x" verts illustrent les résultats simulés dans ce travail lorsque la configuration par défaut est utilisée (cf. section 4.1.1).

Distribution de la probabilité d'occurrence du nombre de DSB/event

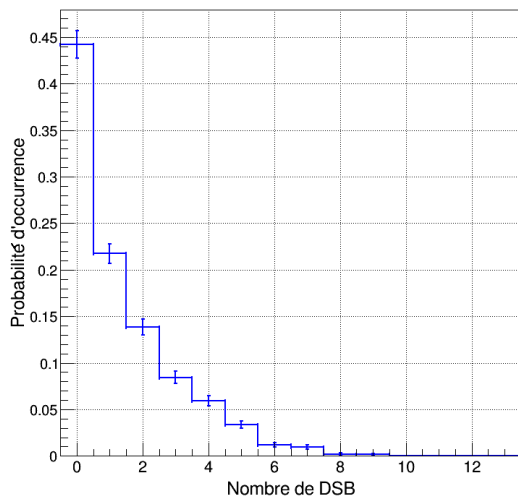
Les nombres moyens de DSB/Gy/Gbp présentées en figure 4.3 et en figure 4.4 sont des moyennes calculées à partir des distributions de la probabilité d'obtenir un certain nombre de DSB par proton incident. La figure 4.5 (cf. page 140) montre six distributions sélectionnées parmi celles générées par FullSim : il s'agit des distributions du nombre de DSB/événements (i.e. particule primaire) obtenues pour des protons incidents de 0,5 ; 0,7 ; 1 ; 3 ; 5 et 10 MeV ce qui correspond à des TEL de 42,6 ; 37,6 ; 28,9 ; 12,8 et 8,7 keV/μm. Il est important de remarquer que toutes les distributions présentent une probabilité de ne détecter aucun DSB nettement plus importante que les autres et, plus l'énergie augmente, plus cette probabilité croit : la probabilité de ne détecter aucun DSB lors du passage d'un proton dans le noyau est d'environ 33% pour 0,5 MeV et de 90% pour 10 MeV. De 0,7 à 10 MeV, les distributions sont continuellement décroissantes et cette décroissance est de plus en plus importante au fur et à mesure que l'énergie augmente. Par exemple, pour 0,7 MeV, la probabilité d'obtenir un unique DSB est de 18% et elle décroît lorsque le nombre de DSB augmente pour atteindre une probabilité nulle d'avoir 14 DSB ou plus. Pour 10 MeV, la probabilité d'avoir un unique DSB est de 10% mais la probabilité d'obtenir 3 DSB ou plus est nulle. Ainsi, la décroissance est donc bien plus importante pour 10 MeV que pour 0,7 MeV. Notons que la probabilité d'obtenir un unique DSB débute à 15% pour 0,5 MeV, augmente pour atteindre environ 23% à 1 MeV et décroît ensuite jusqu'à 10% pour 10 MeV. Remarquons également que, dans l'ensemble, plus l'énergie du proton incident est faible plus la probabilité d'obtenir un grand nombre de DSB augmente.



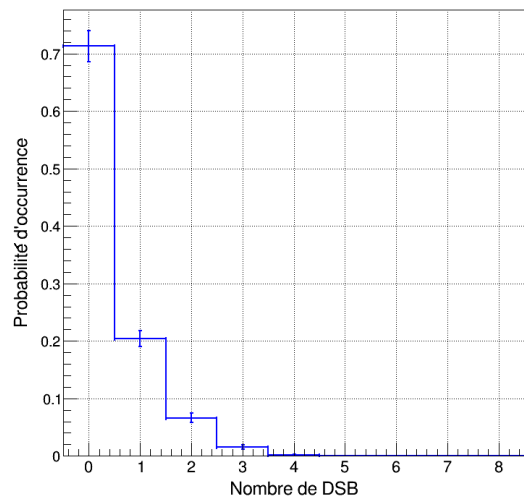
(a) 0,5 MeV



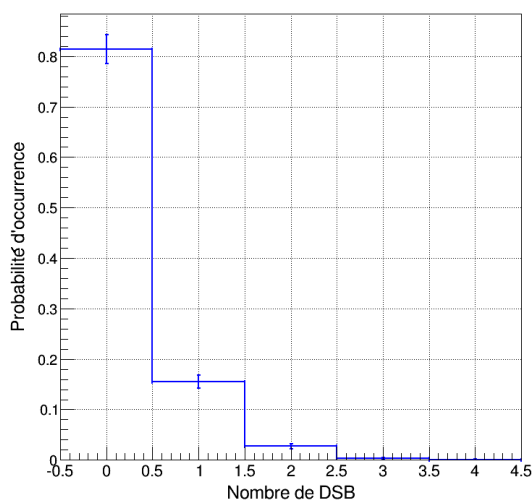
(b) 0,7 MeV



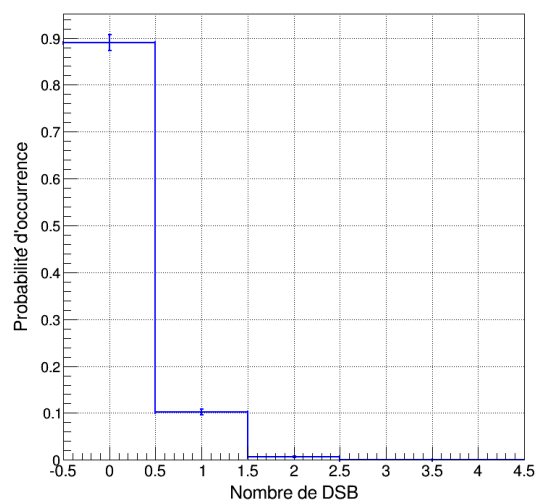
(c) 1 MeV



(d) 3 MeV



(e) 5 MeV



(f) 10 MeV

FIGURE 4.5 – Distributions normalisées du nombre de DSB calculés par particule primaire (event). Les fragments de tailles inférieures à 10000 bp ne sont pas pris en compte et le critère de sélection des SSB_{direct} est le seuil de 17,5 eV (cf. section 3.1.4).

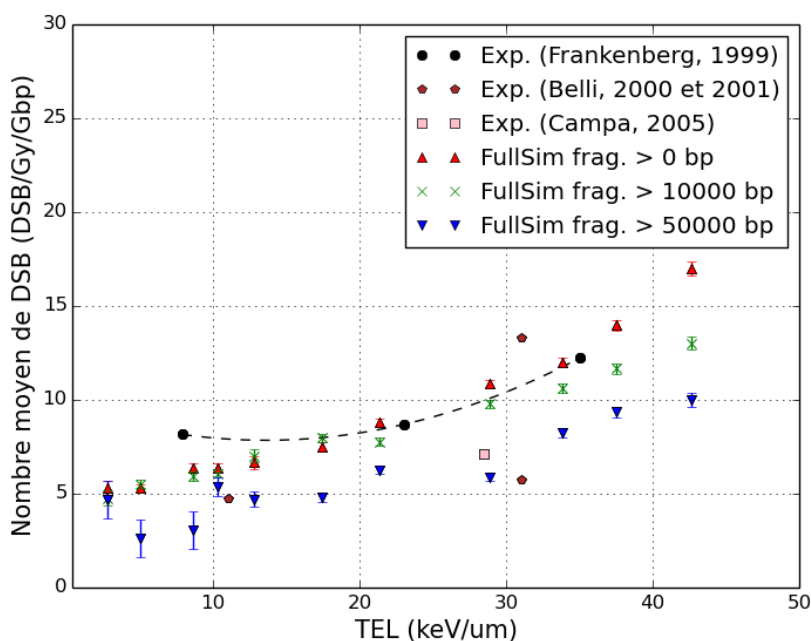


FIGURE 4.6 – Nombre de DSB en fonction de l'énergie du proton incident et de différentes tailles minimales des fragments considérés dans la détection : 0, 10000 et 50000 paires de nucléotides. Le critère de sélection des SSB_{direct} utilisé est le seuil de 17,5 eV (cf. section 3.1.4).

Variation des DSB en fonction de la taille minimale de fragment considérée

La taille minimale de fragment utilisée par défaut est de 10000 bp conformément aux informations fournies par [FRANKENBERG et al., 1999]. Cependant, les expérimentateurs précisent qu'il existe une incertitude associée à ces tailles de fragments minimales et que certains fragments d'ADN de tailles supérieures pourraient également ne pas être détectés. De plus, dans les expériences réalisées par [BELLI et al., 2001, 2000], il est spécifié que la taille minimale de fragment est de 23000 bp. Pour tenter d'estimer l'impact de ces variations, les résultats précédemment générés par notre simulation ont été de nouveau analysés en prenant en compte des tailles de fragments minimales de 0, 10000 et 50000 paires de nucléotides. Notons que lorsque la taille minimale est de 0, tous les fragments sont pris en compte et, donc, tous les DSB simulés sont comptés. Dans les trois cas, le critère de sélection des SSB_{direct} utilisé est toujours le seuil en énergie de 17,5 eV déposés dans le squelette d'un nucléotide. La figure 4.6 montre que le nombre de DSB/Gy/Gbp calculés diminue avec l'augmentation de la taille minimale de fragment choisie. Cet effet s'accroît sensiblement avec l'augmentation du TEL. Ainsi, pour des TEL inférieurs à 10 keV/ μ m, le nombre de DSB/Gy/Gbp est similaire si l'on prend tous les fragments ou uniquement ceux d'une taille supérieure à 10000 bp, compte tenu des incertitudes associées aux résultats. Entre 10 et 20 keV/ μ m, les nombres moyens de DSB/Gy/Gbp calculés avec une taille minimale de 50000 paires de nucléotides sont inférieurs à ceux obtenus avec des tailles minimales de 0 et 10000 paires de nucléotides (bp). Ainsi, entre 10 et 20 keV/ μ m, le fait de passer d'une taille minimale de 0 ou 10000 bp à 50000 bp peut entraîner une diminution du nombre de DSB/Gy/Gbp de 2,5 DSB/Gy/Gbp (50%) au maximum. Pour les TEL supérieurs à 20 keV/ μ m, le nombre moyen de DSB/Gy/Gbp calculés avec une taille minimale de 50000 bp est inférieur à celui obtenu avec une taille minimale de 10000 bp qui est lui-même inférieur au nombre de DSB/Gy/Gbp calculés avec une taille minimale

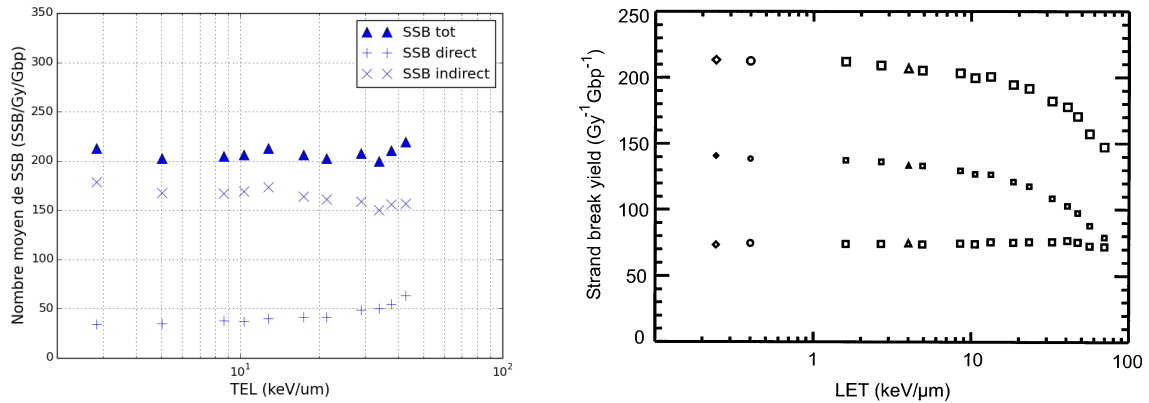
de 0 bp. Ces écarts s'accroissent au fur et à mesure que le TEL augmente pour atteindre des diminutions de 9 DSB/Gy/GBp (24%) entre 0 et 10000 bp et 3 DSB/Gy/GBp (41%) entre 0 et 50000 bp avec un TEL de 42 keV/ μ m. Les résultats expérimentaux de [BELLI et al., 2001, 2000; CAMPA et al., 2005; FRANKENBERG et al., 1999] sont également présentés en figure 4.6 et il apparaît qu'une taille minimale de 50000 bp permet de s'approcher au plus près des résultats de [BELLI et al., 2001, 2000]. Concernant les résultats de [FRANKENBERG et al., 1999], le fait d'utiliser une taille minimale de 0 (compter tous les DSB simulés) semble fournir un meilleur accord qu'avec une taille minimale de 10000 bp.

Nombre de SSB générés

A partir des résultats des simulations précédentes, il est possible d'extraire le nombre de SSB totaux (SSB_{tot}) générés ainsi que de les séparer entre ceux obtenus suite à l'étape physique (SSB_{direct}) et suite à l'étape chimique ($SSB_{indirect}$). Précisons que le calcul des SSB tel qu'il est effectué dans FullSim est indépendant de la taille minimale choisie pour les fragments. Ainsi, les données contenues dans la figure 4.7 sont issues de PARTRAC (cf. figure 4.7b) et de FullSim (avant la phase "DBScan", cf. figure 3.9) dans sa configuration par défaut. Pour les deux types de simulations (PARTRAC et FullSim), le nombre de SSB est présenté en fonction du TEL du proton incident et les quantités de SSB d'origine directe (étape physique) et indirecte (étape chimique) sont indiquées. La figure 4.7 indique ainsi que, pour notre simulation et PARTRAC, le nombre total de SSB est constant jusqu'à environ 1 keV/ μ m. Cependant, ce nombre décroît avec le TEL dans le cas de PARTRAC tandis qu'il reste globalement constant avec notre simulation. Remarquons que les TEL attribués aux protons primaires utilisés dans le cas de PARTRAC sont étonnamment élevés : 90 keV/ μ m pour des protons de 0,5 MeV contre 42 keV/ μ m dans ce travail (cf. tableau 4.1) et la littérature (rapport ICRU 63 et données du NIST). Avec PARTRAC, le nombre de SSB_{direct} est constant et inférieur au nombre de $SSB_{indirect}$ quelque soit le TEL considéré tandis que le nombre de $SSB_{indirect}$ décroît à partir de 10 keV/ μ m. Ces tendances sont différentes avec notre simulation où le nombre de SSB_{direct} augmente avec le TEL tandis que le nombre de $SSB_{indirect}$ diminue légèrement. Remarquons que l'augmentation du nombre de SSB_{direct} s'accroît nettement autour de 40 keV/ μ m. Il apparaît également que le nombre de SSB totaux calculés dans notre simulation est similaire à celui obtenu par PARTRAC pour les TEL inférieurs à 10 keV/ μ m. Cependant, si le nombre de SSB totaux est équivalent dans cette zone entre notre simulation et PARTRAC, ce n'est pas le cas des proportions de SSB_{direct} et $SSB_{indirect}$. En effet, les SSB_{direct} simulés avec FullSim sont inférieurs de 40% à ceux obtenus avec PARTRAC tandis que les $SSB_{indirect}$ calculés avec FullSim sont supérieurs de 20% à ceux de PARTRAC. Au delà de 10 keV/ μ m, le nombre de SSB totaux simulés avec FullSim reste constant voir augmente autour de 40 keV/ μ m tandis que celui issu de PARTRAC décroît de plus en plus fortement. Cette décroissance est également présente au niveau des $SSB_{indirect}$ issus de PARTRAC tandis que ceux de notre simulation semblent décroître légèrement. Toujours pour des TEL supérieurs à 40 keV/ μ m, les SSB_{direct} simulés dans ce travail sont de plus en plus nombreux avec le TEL tandis qu'ils sont constants avec PARTRAC.

Complexité des DSB simulés

La complexité des DSB simulés est à présent examinée, c'est à dire que le nombre de cassures simples composant chaque DSB est étudié. Rappelons qu'un DSB est constitué au minimum de deux cassures simples (SSB) distantes de moins de 10 paires de nucléotides et situées sur des brins opposés de l'ADN. En accord avec cette définition et du fait de



(a) FullSim (Geant4-DNA) avec le critère sélection des SSB_{direct} présent par défaut (seuil de 17,5 eV). (b) PARTRAC [FRIEDLAND et al., 2003] avec une probabilité d'acceptation variant linéairement entre 5 et 37,5 eV.

FIGURE 4.7 – Nombre de SSB calculés en simulant l'irradiation d'un noyau de cellule fibroblaste avec des protons. Les résultats présentés en (a) sont ceux obtenus avec FullSim (configuration par défaut) et les triangles représentent le nombre total de SSB calculés (SSB_{tot}), les "x" les SSB générés lors de l'étape physique (SSB_{direct}) et les "+" les SSB issus de l'étape chimique (SSB_{indirect}). En b), les résultats du code de simulation PARTRAC sont présentés : les grands symboles correspondent aux SSB_{tot}, ceux de taille moyenne aux SSB_{direct} et ceux de petites tailles aux SSB_{indirect}.

l'utilisation de l'algorithme de clusterisation DBSCAN [DOS SANTOS, 2013] intégré dans nos simulations, un DSB peut donc être constitué de plus de deux SSB et ainsi présenter une "complexité" variable. Notons également que certains auteurs établissent une classification des DSB à partir de cette notion de complexité [NIKJOO et al., 2001]. A partir des SSB précédemment extraits, il est possible de déterminer ceux qui constituent des DSB de manière à évaluer la complexité de chaque DSB. Cette information permet de calculer la probabilité d'occurrence des DSB caractérisés par des complexités spécifiques. Ces probabilités sont représentées sous la forme de distributions en annexe C. Les résultats de la figure 4.8 sont issus de ces distributions et décrivent l'évolution en fonction du TEL des DSB dont la complexité est égale à 2, que l'on peut qualifier de "DSB simples", et celle de ceux dont la complexité est supérieure ou égale à 3 ("DSB complexes" selon [NIKJOO et al., 2001]). Les DSB de complexité égale à 2 sont les plus nombreux quelque soit le TEL considéré car ils représentent systématiquement plus de 60% des DSB simulés. Les DSB de complexité supérieure ou égale à 3 ont toujours une fréquence d'apparition inférieure à 40%. Lorsque le TEL augmente, la fréquence relative des DSB de complexité égale à 2 diminue progressivement de 90% pour un TEL de 2,6 keV/μm à 30% pour un TEL de 42,6 keV/μm. Inversement, celle des DSB caractérisés par une complexité supérieure ou égale à 3 augmente de 10% à 32% pour des TEL de 2,6 keV/μm et 42,6 keV/μm respectivement.

4.1.3 Discussion

Cohérence des simulations effectuées

Les résultats présentés en figure 4.3 montrent que lorsqu'on réalise la simulation avec les paramètres spécifiés par défaut alors le nombre de DSB/Gy/Gbp simulés est en bon accord global avec les résultats expérimentaux existant si la dispersion de ces derniers

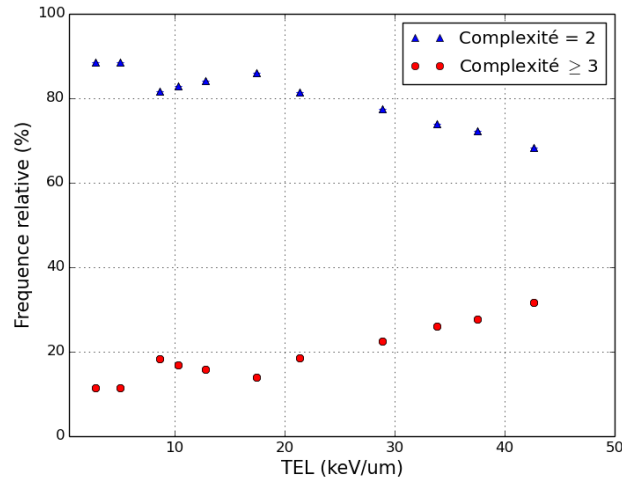


FIGURE 4.8 – Variation du nombre de DSB en fonction de leur complexité et du TEL des protons incidents. Les DSB étant constitués de 2 SSB (complexité = 2) et ceux constitués de 3 SSB au moins (complexité ≥ 3) sont représentés.

est prise en compte. En effet, ces données expérimentales sont issues d'un processus d'analyse complexe et l'incertitude qui leur est associée est importante. Ceci illustrant le fait que les données expérimentales sont peu nombreuses et diverses. Les comparaisons entre les résultats de ce travail et ceux obtenus avec les codes de simulation KURBUC et PARTRAC montrent que les trois codes génèrent des résultats caractérisés par une même tendance : la croissance du nombre de DSB/Gy/Gbp simulés avec le TEL. Précisons que cette constatation est valable dans l'intervalle de TEL étudié (de 2,8 à 42,6 keV/ μm) ainsi que pour des protons. Les résultats issus des trois codes sont du même ordre de grandeur et restent dans l'incertitude globale des données expérimentales. Les différences observées au niveau des valeurs absolues des résultats générés avec les codes étaient toutefois attendues car chaque code possède ses propres spécificités liées aux modèles géométriques, physiques et chimiques qui y sont implémentés. Néanmoins, le critère de sélection des dépôts d'énergie conduisant à un SSB direct, qui est différent dans notre cas et dans celui de PARTRAC, semble avoir une grande importance. Ce point sera à nouveau abordé en section 4.3.2 lorsque différents critères de sélection des SSB_{direct} seront testés. Finalement, l'utilisation de notre chaîne de calcul dans sa configuration par défaut permet de générer des nombres moyens de DSB/Gy/Gbp en bon accord avec les données expérimentales (compte tenu de leur dispersion). Il est ainsi justifié de considérer que les hypothèses et paramètres insérés dans notre chaîne de calcul le sont également. Notons toutefois que l'ensemble de ces paramètres et hypothèses définis par défaut constitue un équilibre adapté à notre simulation mais qui ne peut pas être généralisé. En effet, on ne peut affirmer que la simulation décrit exhaustivement les processus réellement mis en œuvre mais uniquement que l'approximation qui en est faite est cohérente est en accord avec les résultats et les données expérimentales considérées.

Prise en compte de différentes tailles minimales de fragments

La figure 4.6 montre que le fait de considérer des tailles minimales de fragments spécifiques pour chaque expérience permet d'améliorer la cohérence entre les résultats expérimentaux et simulés. Ainsi, la considération d'une taille minimale de fragment de 0 bp au lieu de 10000 bp permet d'obtenir un meilleur accord avec les données de [FRANKENBERG

et al., 1999] tandis que, pour les données de [BELLI et al., 2000], c'est la prise en compte d'une taille minimale de 50000 bp qui occasionne le meilleur accord. Il est intéressant de remarquer que ces tailles minimales de 0 et 50000 bp ne correspondent pas aux tailles minimales spécifiées par les expérimentateurs qui sont de 10000 et 23000 bp respectivement. Ces différences suggèrent que les incertitudes associées aux tailles minimales mentionnées par les expérimentateurs ont une influence significative sur le nombre moyen de DSB/Gy/Gbp simulés. Ajoutons à ce propos que la prise en compte d'une probabilité de détection dépendant de la taille du fragment au lieu d'une limite de détection fixe, telle que 10000 bp par exemple, pourrait être plus réaliste.

La figure 4.6 illustre également le fait que l'influence de la taille minimale des fragments sélectionnée est de plus en plus importante au fur et à mesure que le TEL augmente, comme mentionné dans [FRIEDLAND et al., 2003]. Cette influence croissante signifie que les TEL de plus de 30 keV/ μm occasionnent un nombre important de DSB proches les uns des autres et donc susceptibles de créer des fragments d'ADN trop petits pour être détectés. Ainsi, cette influence se traduit dans nos simulation par une augmentation plus faible du nombre moyen de DSB/Gy/Gbp simulés avec le TEL. Expérimentalement, cela implique qu'un nombre significatif de fragments ne sont pas détectés lorsque des forts TEL sont utilisés.

Complexité des DSB et nombre de SSB simulés

La figure 4.7 permet d'effectuer une comparaison entre les quantités de SSB obtenues avec PARTRAC et FullSim (configuration par défaut). Cette comparaison indique que nous obtenons plus de $SSB_{indirect}$ et moins de SSB_{direct} que PARTRAC mais que ces deux tendances se compensent car les quantités de SSB totaux obtenues dans nos simulations et avec PARTRAC sont similaires. Cependant, le nombre moyen de DSB/Gy/Gbp simulés avec FullSim est presque toujours inférieur à celui obtenu avec PARTRAC, ce qui indique que les SSB générés avec FullSim sont distribués différemment dans l'espace par rapport à ceux issus de PARTRAC. En effet, deux cas sont possibles :

1. Les SSB générés dans nos simulations sont plus rapprochés que ceux issus de PARTRAC. Dans ce cas, les SSB provenant de notre chaîne de calcul forment des DSB plus complexes et le nombre est inférieurs à ceux calculés avec PARTRAC en raison de cette complexité accrue.
2. Les SSB générés par FullSim sont plus éloignés et moins de DSB sont formés. En effet, rappelons qu'au moins deux SSB doivent être situés sur des brins d'ADN opposés et distants de moins de 10 paires de nucléotides pour former un DSB.

A ce propos, les variations de la complexité des DSB simulés dans notre travail montrent que la proportion de DSB complexes s'accroît lorsque le TEL augmente (cf. figure 4.8 et annexe C) et ce résultat est cohérent avec plusieurs simulations précédemment effectuées et décrites dans la littérature [FRIEDLAND et al., 2010; NIKJOO et LINDBORG, 2010; PATER et al., 2016]. Étant donné que cet accroissement de la complexité avec le TEL est corrélé à un meilleur accord avec les résultats de PARTRAC (cf. figure 4.4), il est raisonnable de supposer que les écarts de résultats constatés à de plus bas TEL vont dans le sens du cas numéro 2, c'est à dire que les SSB générés par notre simulation sont plus éloignés les uns des autres que ceux issus de la simulation effectuée avec PARTRAC. Toutefois, seule une comparaison de la complexité des DSB générés dans notre simulation avec celle de PARTRAC permettrait d'en avoir la certitude.

Notons également que l'écart entre les résultats de notre chaîne de calcul et ceux de PARTRAC pourrait, entre autres, provenir de la manière dont le "scavenging" des radi-

caux OH[•] est simulé durant l'étape chimique. En effet, l'effet des "scavengers" est pris en compte de manière discrète dans PARTRAC et pas dans notre cas. Concrètement, cela signifie que PARTRAC retire un certain nombre de radicaux OH[•] lors de chaque pas en temps effectué durant la simulation de l'étape chimique [FRIEDLAND et al., 2003]. Cette dernière étape dure environ 10 ns dans PARTRAC. Dans ce travail, nous procédons de manière plus globale en limitant la durée totale de l'étape chimique à 2,5 ns et ne retirant aucun radical OH[•] lors des pas en temps. Ces différentes prises en compte des "scavengers" pourraient être à l'origine des écarts observés en influant sur la distribution spatiale des SSB générés lors de l'étape chimique qui s'avèrent être majoritaires. L'influence de la durée limitée de l'étape chimique prise en compte dans nos simulations sera étudiée plus en détail en section 4.3.3.

Importance de la probabilité de ne pas causer de dommages à l'ADN

Les distributions de la figure 4.5 sont à l'origine des nombres moyens de DSB/Gy/Gbp calculés précédemment et introduits en figure 4.3 et en figure 4.4. Sans surprise, ces distributions montrent bien qu'une augmentation du TEL entraîne une augmentation de la probabilité d'obtenir plus de 2 DSB par particule primaire (ou événement) induisant mécaniquement une hausse de la valeur moyenne calculée à partir de la distribution. Ceci explique la croissance du nombre moyen de DSB/Gy/Gbp avec le TEL constatée en figure 4.3 et 4.4. Les distributions illustrent également le fait que la probabilité de ne pas générer de DSB lors du passage d'un proton primaire dans le noyau cellulaire est toujours largement dominante quelque soit le TEL considéré. Cela a également été observé lors de mesures effectuées dans le cadre de la nanodosimétrie où la probabilité de ne détecter aucune ionisation dans le détecteur de taille nanométrique est dominante [BANTSAR, 2012; BANTSAR et al., 2009; PSZONA et al., 2000]. Toutefois, cet effet est contrebalancé dans nos simulations par la nécessité de prendre en compte de nombreux protons primaires pour arriver à déposer une dose de 1 Gy dans le noyau (cf. équation 4.1, section 4.1.1) expliquant ainsi que le nombre moyen de DSB/Gy/Gbp simulés ne soit jamais nul.

4.2 Influence des sections efficaces de l'ADN

Les simulations effectuées dans le cadre de la section 4.1 ont permis de vérifier l'ajustement des paramètres associés à la simulation ainsi que la cohérence de l'ensemble du calcul vis à vis de résultats expérimentaux et théoriques. La méthode de calcul étant fiable, le système de gestion des matériaux introduit dans le chapitre précédent a été intégré pour étudier l'influence de la prise en compte l'ADN en tant que matériau composite lors du transport de particule. Des informations plus détaillées concernant ce matériau composite sont fournies dans le chapitre précédent (cf. section 3.1.1). Cette section présente les résultats obtenus et les compare avec ceux précédemment générés lorsque l'eau liquide était le seul matériau considéré (cf. la section 4.1).

4.2.1 Configuration des simulations

Deux ensembles de simulations sont effectués dans cette section, le premier a pour but d'étudier l'influence de l'introduction des sections efficaces de l'ADN à l'échelle de quelques paires de nucléotides tandis que le second vise à estimer l'impact associé à l'introduction d'un matériau composite représentatif de l'ADN sur des simulations menées à l'échelle de noyaux entiers. L'étendue spatialement réduite (10 paires de nucléotides) des

simulations du premier ensemble permet d'évaluer les effets du nouveau matériau sur les électrons secondaires simulés. Cette étude est ensuite étendue de manière à considérer l'influence du nouveau matériau ADN sur les électrons secondaires, les protons primaires et à l'échelle d'un noyau cellulaire.

Simulations dans 10 paires de nucléotides

L'objectif de ce premier ensemble de simulations est d'estimer l'influence d'un matériau représentatif de l'ADN à petite échelle (quelques paires de nucléotides). Les simulations menées à cet effet sont similaires à celles mentionnées en section 3.1.2. Pour rappel, la géométrie est constituée d'un petit morceau d'ADN de 10 paires de nucléotides qui est représenté en figure 3.3. Dans cette section, le morceau d'ADN est successivement rempli d'eau liquide puis du matériau composite ADN ($\rho = 1,56 \text{ g/cm}^3$) introduit en section 3.1.1 et, enfin, du même matériau mais avec une masse volumique ρ de 1 g/cm^3 . Cette variation de la masse volumique du matériau composite vise à estimer séparément l'influence sur nos simulations des sections efficaces de l'ADN et de la masse volumique du matériau composite. En effet, les sections efficaces utilisées lors du transport des particules dans l'ADN sont systématiquement multipliées par la masse volumique du matériau utilisé pour l'ADN (cf. section 3.1.1) rendant ainsi indistinguable l'influence des sections efficaces seules. L'utilisation d'une masse volumique de 1 g/cm^3 permet ainsi de se concentrer uniquement sur les sections efficaces. Précisons que la masse volumique du matériau utilisé dans l'ADN est étroitement liée à la modélisation géométrique de l'ADN qui détermine son volume. Ainsi, la masse volumique du matériau composite ADN ($\rho = 1,56 \text{ g/cm}^3$) est susceptible de varier lors d'éventuelles et futures modifications de la modélisation géométrique. Des électrons sont utilisés en tant que particules primaires et leurs énergies cinétiques initiales vont de 50 à 800 eV. La source des particules primaires se situe sur une surface cylindrique englobant les 10 paires de nucléotides et sont systématiquement envoyés en direction de ces dernières. En terme de résultats, les ionisations générées dans le cylindre sont enregistrées pour chaque électron primaire ainsi que pour chaque matériau et volume constitutif de l'ADN. Au final et à partir des ionisations enregistrées, un nombre moyen d'ionisations par électron primaire est obtenu pour chaque matériau et pour chaque volume : il s'agit du M1. L'attribution des ionisations à un composant de l'ADN est effectuée avec la méthode "volumique" précédemment choisie en section 3.1.2.

Simulations dans un noyau de cellule fibroblaste

La configuration du second ensemble de simulations effectuées dans cette section est identique à celle utilisée en section 4.1, la seule différence se situant au niveau du matériau pris en compte lors du transport de particules. En effet, les simulations de la section 4.1 ont eu lieu dans de l'eau liquide uniquement tandis que les simulations de cette section prennent en compte l'ADN en tant que matériau composite. Les caractéristiques principales des simulations menées dans cette section sont résumées ci-dessous :

- La géométrie prise en compte est la même qu'en section 4.1, soit un noyau de cellule fibroblaste contenant un génome humain mâle.
- Les particules primaires sont des protons tirés en direction du noyau selon la configuration décrite en figure 4.1. Les énergies de ces protons mono-énergétiques vont de 0,5 MeV à 20 MeV ce qui correspond à des TEL allant de 2,8 à 42,6 keV/ μm .

- L'ADN est considéré comme un matériau composite constitué d'eau, 2-désoxyribose, phosphate, adénine, guanine, thymine et cytosine. La densité de ce matériau est de $1,56 \text{ g/cm}^3$.

Sauf indiqué autrement, les paramètres associés à ces simulations sont ceux utilisés par défaut dans FullSim, c'est à dire les mêmes que ceux utilisés en section 4.1. Le fait d'utiliser des paramètres similaires facilite la comparaison entre les résultats obtenus avec l'eau liquide comme unique matériau et ceux calculés avec le matériau composite ADN. Pour rappel, le critère de sélection des $\text{SSB}_{\text{direct}}$ utilisé est un seuil de 17,5 eV appliqué à l'énergie déposée dans le squelette de chaque nucléotide et l'attribution d'un composant de l'ADN à une interaction est effectuée de manière volumique conformément au choix effectué en section 3.1.2. Après chaque simulation, les résultats sont produits et analysés comme en section 4.1. Les résultats finaux sont donc présentés par Gy et par Gbp en fonction du TEL de la particule primaire incidente.

4.2.2 Résultats : influence du matériau composite ADN sur le nombre d'ionisations à l'échelle de 10 paires de nucléotides

Nous comparons ensuite les résultats obtenus avec le matériau composite ADN ($\rho = 1,56 \text{ g/cm}^3$) aux résultats obtenus avec l'eau liquide ($\rho = 1 \text{ g/cm}^3$) en figure 4.9. Pour ces calculs, le choix a été fait de ne considérer qu'un seul type d'attribution pour les ionisations : l'attribution volumique. En effet, c'était le seul choix possible dans ce contexte car, avec l'eau liquide, un dépôt d'énergie ne peut-être attribué qu'à un volume géométrique. Chaque ionisation est ainsi attribuée à un composant de l'ADN et bien que l'eau liquide soit présente parmi ces composants, elle n'est pas représentée sur la figure. Les courbes labellisées "... (ADN)" en figure 4.9 sont donc identiques à celles de la figure 3.4 du chapitre 3 en considérant une attribution volumique des ionisations. Les variations du M1 en fonction de l'énergie et pour chaque composant suivent la même tendance que celle précédemment remarquée en figure 3.4 : croissance jusqu'à 180 eV puis décroissance. Les valeurs du M1 les plus importantes sont obtenues pour le désoxyribose et ces valeurs atteignent un maximum 0,38 ionisations/10bp pour des électrons primaires de 180 eV. Dans l'ensemble, il apparaît que le fait de remplacer l'eau liquide par le matériau composite ADN augmente le nombre moyen d'ionisations enregistrées quelque soit le matériau et l'énergie considérée.

La figure 4.10 présente des valeurs de M1 obtenues à partir de simulations similaires à celles utilisées pour générer les résultats introduits en figure 4.9. La seule différence entre les deux jeux de simulations se situe au niveau des masses volumiques associées aux matériaux remplissant l'ADN. En effet, en figure 4.9, l'ADN est rempli avec de l'eau liquide puis avec le matériau composite représentatif avec une masse volumique de $1,56 \text{ g/cm}^3$ tandis que, pour les simulations associées à la figure 4.10, la masse volumique du matériau composite représentatif de l'ADN a été modifiée pour être équivalente à celle de l'eau liquide : 1 g/cm^3 . Comme en figure 4.9, les ionisations sont attribuées à un composant de l'ADN de manière volumique. Les valeurs de M1 calculées avec de l'eau liquide et le matériau composite ($\rho = 1 \text{ g/cm}^3$) sont très proches pour un même composant de l'ADN. Pour une énergie et un composant donné, le fait de passer l'eau liquide au matériau composite ($\rho = 1 \text{ g/cm}^3$) occasionne une différence toujours inférieure à 2%. Il est cependant intéressant de remarquer que, pour des énergies inférieures à 180 eV, le nombre moyen d'ionisations calculé avec le matériau composite est plus important que celui calculé avec l'eau liquide. A l'inverse, au delà de 180 eV, les M1 les plus importants sont obtenus dans de l'eau liquide et cet effet est particulièrement visible pour les composants désoxyribose

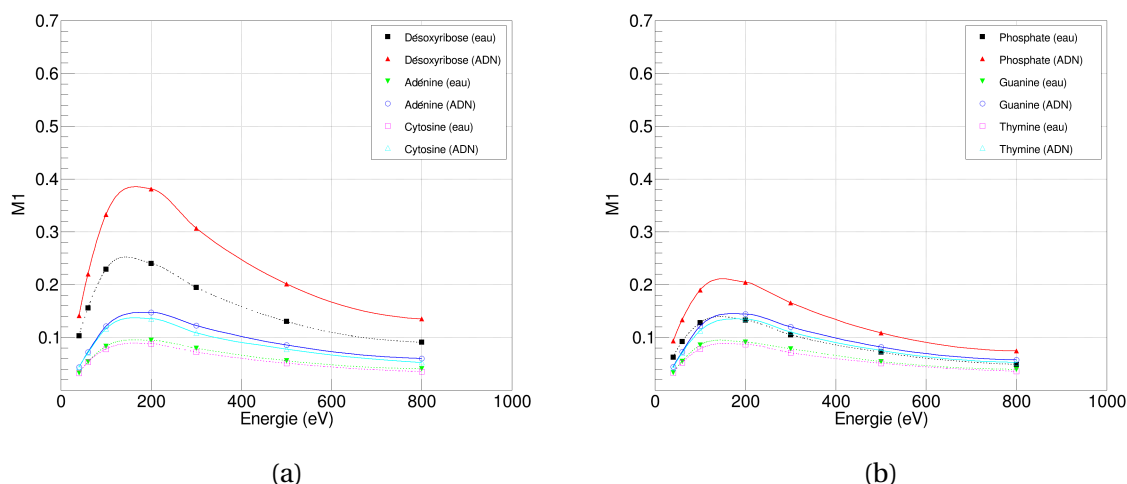


FIGURE 4.9 – Nombre moyen d’ionisations enregistrées dans 10 paires de nucléotides en fonction de l’énergie des électrons primaires et du matériaux pris en compte [(a) et (b)] : un des composants de l’ADN ou l’eau liquide. L’attribution des ionisations à un composant de l’ADN est volumique. La masse volumique ρ du matériau composite ADN est de $1,56 \text{ g/cm}^3$.

et phosphate.

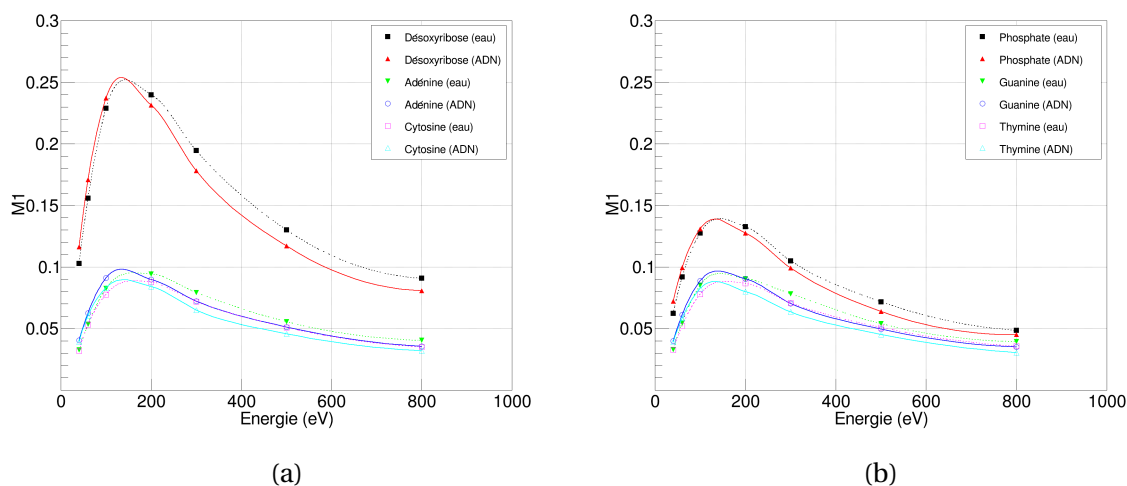


FIGURE 4.10 – Nombre moyen d’ionisations enregistrées dans 10 paires de nucléotides en fonction de l’énergie des électrons primaires et du matériau utilisé pour remplir l’ADN [(a) et (b)] : l’eau liquide ou le matériau composite ADN. La masse volumique ρ du matériau composite ADN est, dans ce cas précis, de 1 g/cm^3 .

4.2.3 Résultats : influence du matériau composite ADN à l’échelle du noyau de cellule fibroblaste

L’irradiation du noyau de cellule fibroblaste a été simulée à nouveau avec des protons incidents de TEL allant de 2,8 à 42,6 keV/ μm (protons mono-énergétiques) mais, cette fois, l’ADN contenu dans ce noyau n’est pas rempli d’eau liquide mais du matériau composite représentatif introduit en section 3.1.1. Le nombre de DSB/Gy/Gbp calculé à partir

de ces simulations est présenté en figure 4.11 conjointement avec le nombre de DSB/Gy/Gbp précédemment obtenu lorsque l'eau liquide était l'unique matériau utilisé. La comparaison de ces deux courbes montre clairement que la prise en compte du matériau composite ADN entraîne une augmentation du nombre de DSB/Gy/Gbp. Cette augmentation oscille entre 2 et 5 DSB/Gy/Gbp pour tous les TEL considérés. L'augmentation la plus faible est obtenue pour des protons incidents avec un TEL de 2,8 keV/μm, il s'agit alors d'une augmentation de 30%. L'évolution du nombre de DSB/Gy/Gbp en fonction de l'énergie est similaire lorsque la géométrie de l'ADN est remplie d'eau liquide ou lorsqu'elle contient le matériau composite ADN : le nombre de DSB/Gy/Gbp simulés augmente avec le TEL dans les deux cas.

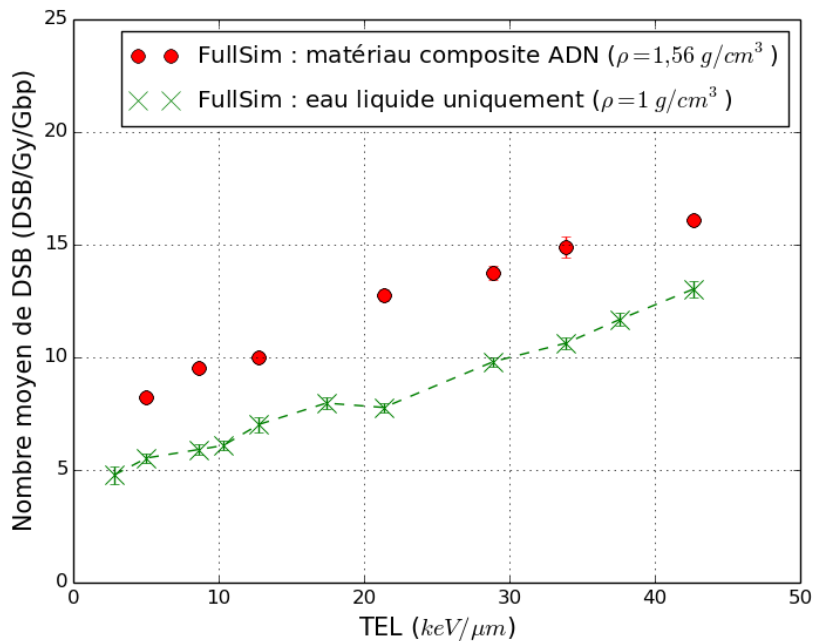


FIGURE 4.11 – Nombre moyen de DSB/Gy/Gbp générés par des protons de différentes énergies dans un noyau de cellule fibroblaste contenant de l'ADN. L'ADN est constitué d'eau liquide (points verts) puis du matériau composite ADN (points rouges) de masse volumique : $\rho = 1,56 \text{ g/cm}^3$. Les simulations réalisées avec le matériau composite utilisent le système de gestion des matériaux implémenté dans Geant4.LDRI.01 pour prendre en compte les sections efficaces des composants de l'ADN. Les valeurs des paramètres des simulations sont celles attribuées par défaut (taille minimale de fragment de 10000 bp et seuil de 17,5 eV dans le backbone pour les SSB_{direct}).

De la même manière qu'en figure 4.7, les résultats issus de simulations précédemment menées avec le matériau composite ADN ont été analysés pour en extraire les SSB. Ces SSB (SSB_{tot}) ont ensuite été catégorisés en SSB_{direct} (étape physique) et $SSB_{indirect}$ (étape chimique) tel qu'illustré en figure 4.12. Sur cette figure, le nombre de SSB est présenté en tant que $SSB/Gy/Gbp$ et en fonction du TEL des protons primaires utilisés ainsi que du matériau utilisé dans la géométrie (matériau composite ADN ou eau liquide). Ainsi, la figure 4.12 montre que l'introduction du matériau composite ADN induit une augmentation du nombre de SSB_{tot} pour tous les TEL représentés. Cette augmentation est, au minimum, de 5% pour 42,6 keV/μm et, au maximum, de 16% pour 5 keV/μm. Que cela soit avec l'eau liquide ou le matériau composite ADN, la quantité de $SSB_{indirect}$ calculée est toujours supérieure à celle de SSB_{direct} . De même, quelque soit le matériau utilisé, le nombre SSB_{direct} augmente pour des TEL supérieurs à 10 keV/μm tandis que le

nombre de $SSB_{indirect}$ diminue légèrement. De façon notable, le fait d'utiliser le matériau composite ADN augmente le nombre de SSB_{direct} de 50% en moyenne pour tous les TEL représentés tandis que, inversement, le nombre de $SSB_{indirect}$ diminue de 10%.

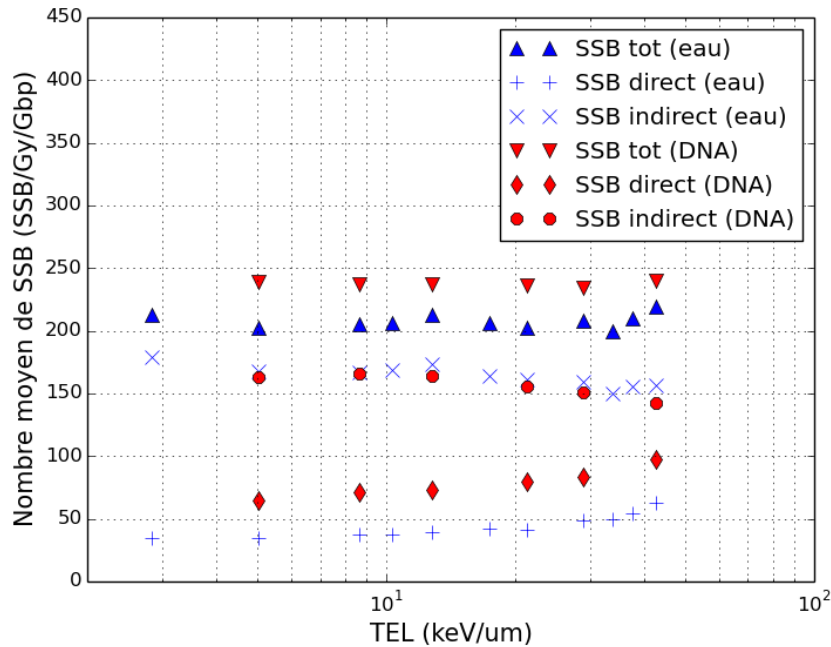


FIGURE 4.12 – Nombre moyen de SSB totaux (directs et indirects) calculés dans ce travail avec un noyau de cellule fibroblaste. Deux cas sont considérés : l'ADN est rempli d'eau liquide (points bleus, $\rho = 1 \text{ g/cm}^3$) et l'ADN est constitué du matériau composite représentatif (points rouges, $\rho = 1,56 \text{ g/cm}^3$). Les simulations sont effectuées avec les valeurs présentes par défaut dans notre chaîne de calcul (cf. section 3.1.4) et les particules primaires sont des protons.

La figure 4.13 (cf. page 4.13) montre le nombre de DSB/Gy/Gbp calculés en fonction de la masse volumique du matériau composite ADN lors de la simulation de l'irradiation d'un noyau de fibroblaste humain avec des protons primaires de 1 MeV (28,9 keV/ μm). Dans un premier temps, l'ADN contenu dans le noyau est rempli d'eau liquide puis, ensuite, du matériau composite ADN dont la masse volumique vaut successivement : 1 g/cm^3 ($\rho_{composite} = \rho_{eau-liquide}$), $1,248 \text{ g/cm}^3$ (-20%), $1,56 \text{ g/cm}^3$ et $1,812 \text{ g/cm}^3$ (+20%). Dans toutes ces simulations, il n'y a pas de taille minimale de fragment pour que tous les DSB simulés soient enregistrés. Les résultats de la figure 4.13 montrent que le nombre de DSB/Gy/Gbp simulés dépend fortement de la masse volumique du matériau associé à l'ADN : le nombre de DSB/Gy/Gbp augmente avec cette dernière. En effet, pour un matériau composite avec $\rho = 1 \text{ g/cm}^3$ le résultat est de 10,8 DSB/Gy/Gbp contre 18,7 DSB/Gy/Gbp pour un matériau composite avec $\rho = 1,812 \text{ g/cm}^3$, ce qui représente une augmentation de 73%. La comparaison des résultats obtenus avec de l'eau liquide et le matériau composite ADN de même densité ($\rho = 1 \text{ g/cm}^3$) montre que ces derniers sont extrêmement proches et peuvent être considérés comme identiques compte tenu de l'incertitude statistique qui leur est associé.

Le tableau 4.2 (cf. page 152) présente le nombre moyen de DSB/Gy/Gbp obtenu lorsque l'eau liquide ou le matériau composite ADN ($\rho = 1 \text{ g/cm}^3$) sont utilisés pour transporter les particules dans l'ADN lors de l'étape physique. Les résultats du tableau 4.2 dépendent également du critère de sélection des SSB_{direct} choisi : seuil de 17,5 eV ou probabilité d'acceptation variant linéairement en fonction de l'énergie déposée. Les résul-

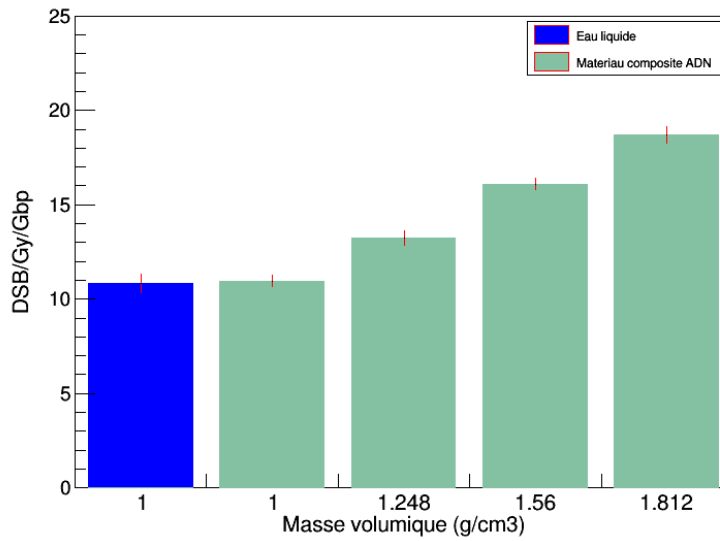


FIGURE 4.13 – Nombre de DSB/Gy/Gbp en fonction de la densité du matériau composite ADN, dans le cas d’une irradiation du noyau de fibroblaste avec des protons primaires de 1 MeV. La configuration de FullSim est celle définie par défaut à l’exception de la taille minimale de fragment qui est de 0 pour que tous les DSB soient comptabilisés.

tats montrent qu’avec le seuil de 17,5 eV (configuration par défaut) le nombre moyen de DSB/Gy/Gbp est constant avec l’eau liquide et le matériau composite ADN : la différence ne dépasse pas les +2%. A l’inverse, lorsque la probabilité linéaire est utilisée en tant que critère de sélection alors l’utilisation du matériau composite ADN ($\rho = 1 \text{ g/cm}^3$) induit une diminution de -16,6% du nombre moyen de DSB/Gy/Gbp simulés vis à vis de celui obtenu avec de l’eau liquide.

TABLEAU 4.2 – Nombre moyen de DSB/Gy/Gbp simulés en fonction du matériau utilisé pour l’ADN (eau liquide ou matériau composite ADN, tout deux avec $\rho = 1 \text{ g/cm}^3$) et du critère de sélection des SSB_{direct} choisi. Les particules primaires utilisées sont des protons de 1 MeV et la cible est l’ADN contenu dans un noyau de cellule fibroblaste. Les résultats présentés sont obtenus lorsque tous les fragments sont comptabilisés (pas de taille minimale).

	Seuil de 17,5 eV	Proba. lin. de 5 à 37,5 eV
Eau liquide	10,8	16,3
Composite ADN	11,0	13,6

4.2.4 Discussion

Prise en compte du matériau composite représentatif de l’ADN

Les résultats de la figure 4.11 montrent que l’utilisation du matériau composite ADN à l’échelle du noyau de cellule fibroblaste augmente le nombre de DSB simulés. Ce dernier surestime alors les données expérimentales et cela est dû au fait que les valeurs choisies pour les paramètres de notre simulation étaient précisément sélectionnées pour équilibrer une simulation avec de l’eau liquide comme unique matériau. Dans ce contexte, l’introduction du matériau composite a une influence significative en augmentant le nombre de DSB/Gy/Gbp simulés et le maintien de la cohérence de la simulation nécessiterait un

ré-équilibrage des paramètres. Un tel ré-équilibrage permettrait à la simulation de fournir des résultats similaires à ceux précédemment obtenus avec l'eau liquide uniquement. Parmi les paramètres ayant une grande influence sur les résultats, le plus significatif serait la méthode de sélection des SSB_{direct} . Le fait d'introduire le matériau composite représentatif de l'ADN lors du transport des particules induit une augmentation significative du nombre d'ionisations ayant lieu dans 10 paires de nucléotides comme l'illustre la figure 4.9. Cette augmentation signifie que les électrons secondaires¹ simulés interagissent plus souvent dans l'ADN lorsque ce dernier contient le matériau composite. Comme les électrons secondaires sont très majoritairement présents dans la simulation, leurs plus nombreuses interactions dans l'ADN entraînent une hausse significative de la probabilité d'endommager ce dernier (SSB_{direct}). Et ceci est d'autant plus pertinent que l'augmentation du nombre d'ionisations représenté en figure 4.9 est la plus significative pour la désoxyribose et le phosphate qui sont précisément les volumes à partir desquels les SSB sont déterminés (le squelette du nucléotide). Le fait que les résultats obtenus à l'échelle du noyau montrent une augmentation de nombre moyen de DSB/Gy/Gbp n'est, à ce titre, pas surprenant.

La figure 4.12 indique que l'introduction du matériau composite ADN a pour conséquence une augmentation du nombre total de SSB calculés. Toutefois, il est intéressant de constater que cette augmentation ne se traduit pas identiquement en terme de SSB_{direct} et $SSB_{indirect}$. En effet, le matériau composite ADN n'est utilisé que lors de la simulation de l'étape physique et son introduction impacte donc principalement les SSB_{direct} . Ainsi, si les SSB_{direct} calculés sont naturellement plus nombreux, ce n'est pas le cas des $SSB_{indirect}$ qui sont moins nombreux comparés à ceux obtenus avec un ADN rempli d'eau liquide uniquement. L'augmentation des SSB_{direct} est toutefois plus significative que la diminution des $SSB_{indirect}$ expliquant ainsi l'augmentation globale du nombre de SSB totaux. La diminution du nombre de $SSB_{indirect}$ provient du fait que, dans notre simulation, toute molécule de 2-désoryribose ionisée lors de l'étape physique est retirée de l'étape chimique de manière à ne pas être endommagée deux fois. Comme une augmentation des SSB_{direct} implique une augmentation du nombre de molécules de désoxyribose endommagées, ces dernières sont moins nombreuses lors de la simulation de l'étape chimique et la probabilité de générer des $SSB_{indirect}$ diminue en conséquence provoquant ainsi la baisse du nombre de $SSB_{indirect}$ observé en figure 4.12.

Influence de la densité

L'augmentation du nombre d'ionisations, constatée en figure 4.9, lors de l'introduction du matériau composite ADN provient de deux éléments :

- L'utilisation des sections efficaces de l'ADN à la place de celles de l'eau liquide.
- Le fait que la masse volumique du matériau composite soit de $1,56 \text{ g/cm}^3$ alors que celle de l'eau est de 1 g/cm^3 .

Les résultats de la figure 4.13 permettent d'évaluer l'impact de chacun des éléments mentionnés et il apparaît que le fait de faire varier la densité du matériau composite ADN de $\pm 20\%$ ($1,56 + 20\% = 1,872$ et $1,56 - 20\% = 1,248$) induit des différences significatives concernant le nombre de DSB/Gy/Gbp calculés : -18% de DSB en moins pour une densité de -20%, 16 DSB avec $\rho = 1,56 \text{ g/cm}^3$ et +19% de DSB pour une densité de +20%. Ceci montre que l'impact de la masse volumique du matériau composite sur le nombre de

1. Un électron secondaire est un électron mis en mouvement par une particule primaire (proton ou ion alpha dans ce travail).

DSB/Gy/Gbp est très significatif. Les résultats obtenus en considérant un matériau composite ADN ayant une masse volumique de 1 g/cm^3 et l'eau liquide permettent d'étudier uniquement l'influence des sections efficaces des différents matériaux. Il apparaît alors que le nombre total de DSB/Gy/Gbp simulés est identique avec l'eau liquide et le matériau composite ADN ($\rho = 1 \text{ g/cm}^3$). Cette constatation effectuée à l'échelle du noyau cellulaire peut également être faite à l'échelle de 10 paires de nucléotides car des résultats similaires ont été observés dans ce travail (cf. figure 4.10) et dans le travail de thèse de [BUG, 2014] (chapitre 5, page 188). Ces résultats montrent que les électrons secondaires sont influencés par l'utilisation des sections efficaces de l'ADN mais la mise en perspective de ces résultats avec ceux de la figure 4.9 démontre que l'influence de la masse volumique du matériau utilisée est largement dominante. Ainsi, le comportement des électrons secondaires simulés dans le noyau cellulaire, lorsque le matériau composite ADN ($\rho = 1 \text{ g/cm}^3$) est présent, suit les mêmes tendances que celles observées en figure 4.10 mais ces dernières sont trop faibles pour modifier significativement le nombre de DSB/Gy/Gbp simulés. La prise en compte de la masse volumique du matériau composite ADN change cela en provoquant une hausse des ionisations dans l'ADN. Cette hausse fait en retour apparaître l'augmentation du nombre de DSB/Gy/Gbp constaté en figure 4.11. Précisons toutefois que la prise en compte du matériau composite ADN n'est réaliste que si elle est comprise à la fois les sections efficaces de l'ADN et le changement de masse volumique associé.

Il est intéressant de remarquer que l'usage de la probabilité d'acceptation linéaire en tant que critère de sélection des SSB_{direct} fait apparaître une différence entre les résultats obtenus avec l'eau liquide et le matériau composite ADN ($\rho = 1 \text{ g/cm}^3$). En effet, le tableau 4.2 montre que l'utilisation du matériau composite ADN ($\rho = 1 \text{ g/cm}^3$) conjointement avec la probabilité d'acceptation linéaire induit une diminution du nombre moyen de DSB/Gy/Gbp calculé. Et cette diminution apparaît malgré le fait que les sections efficaces utilisées pour l'ADN soient bien plus élevées que celles utilisées dans l'eau liquide (cf. annexe A et B). Pour expliquer cela, il est nécessaire de considérer les couches électroniques des composants du matériau composite ADN. En effet, les couches des composants de l'ADN sont plus nombreuses que celles de l'eau liquide et également plus profondes. Cela a pour conséquence que les interactions ayant lieu avec le matériau composite ADN occasionnent des dépôts d'énergie plus importants que celles impliquant uniquement l'eau liquide. Ainsi, les sections efficaces plus élevées dans l'ADN (i.e. dans le matériau composite ADN) augmentent bien la probabilité qu'une particule incidente interagisse dans l'ADN mais cette même particule a également une probabilité bien plus élevée de déposer une énergie importante de le milieu et, donc, d'être retirée rapidement de la simulation (thermalisation). Cet effet peut également être observé à l'échelle de quelques paires de nucléotides en figure 4.10 où le nombre d'ionisations enregistrées est plus élevé avec l'eau liquide que le matériau composite ADN ($\rho = 1 \text{ g/cm}^3$) et dans la littérature [BUG et al., 2012]. A l'échelle du noyau cellulaire et des DSB/Gy/Gbp, l'utilisation de la probabilité d'acceptation linéaire implique une probabilité non nulle que les dépôts d'énergie compris entre 5 et 17,5 eV génèrent des SSB_{direct} . En raison de la structure moléculaire de l'eau liquide et des composants de l'ADN, la probabilité d'avoir des dépôts d'énergie compris entre ces valeurs est plus importante dans l'eau liquide que dans l'ADN (cf. le tableau 3.2 en section 3.1.1). Ceci expliquant que le nombre moyen de DSB/Gy/Gbp simulé soit plus important avec l'eau liquide que le matériau composite ADN lorsque la probabilité d'acceptation linéaire est utilisée.

4.3 Sensibilité de la simulation

L'objectif de cette section est d'estimer la sensibilité des simulations menées dans ce travail à trois paramètres distincts :

1. Le critère de sélection utilisé pour sélectionner les dommages directs qui deviendront des SSB_{direct}, par défaut ce critère est un seuil de 17,5 eV appliqué à l'énergie déposée dans le squelette de chaque nucléotide ("backbone"). Si l'énergie déposée dépasse ce seuil alors un SSB_{direct} est comptabilisé.
2. La durée simulée de l'étape chimique qui est de 2,5 ns par défaut.
3. L'influence du découpage de la chromatine en voxels isolés (50 nm de côté) lors de la simulation des étapes physique et chimique.

Les valeurs prises par défaut par ces paramètres constituent des hypothèses dont l'influence sur les résultats obtenus doit être estimée. A ce titre, les simulations effectuées dans cette section ont pour objectif de tester ces hypothèses et d'estimer leur impact potentiel sur le nombre de DSB/Gy/Gbp calculés dans ce travail.

4.3.1 Configuration des simulations

Les simulations menées pour générer les résultats présentés dans cette section sont regroupées en trois ensembles correspondant aux trois paramètres précédemment mentionnés. Hormis ces paramètres, les simulations sont effectuées avec la configuration par défaut, soit pour rappel :

1. Toute ionisation ou excitation (électronique ou vibrationnelle) ayant lieu dans une molécule de phosphate, de désoxyribose ou dans leurs couches d'hydratation respectives sera prise en compte pour le calcul des dommages directs.
2. Condition sur l'apparition d'un SSB direct : seuil en énergie de 17,5 eV appliqué à l'énergie déposée dans le squelette de chaque nucléotide. Si cette énergie déposée dépasse ce seuil alors un SSB_{direct} est enregistré.
3. Condition sur l'apparition d'un SSB indirect : toute réaction entre un radical OH• et une molécule de désoxyribose a 42% de chances d'entraîner un SSB indirect.
4. Une cassure double brin (DSB) est formée de, au minimum, deux SSB situés sur des brins opposés et séparés entre eux par moins de 10 paires de nucléotides.
5. Le temps simulé lors de l'étape chimique est de 2,5 ns.
6. La présence des molécules de phosphate n'est pas prise en compte lors de la simulation chimique.
7. Durant l'étape chimique, le complexe de protéine histone est représenté par une sphère de 2,4 nm de rayon qui absorbe tout ce qui entre en contact avec elle.

Notons que les DSB à l'origine de fragments d'ADN de tailles inférieures à 10000 paires de nucléotides ne sont pas comptabilisés et que le seul matériau utilisé est l'eau liquide.

Sélection des dommages directs

Pour estimer l'influence du critère de sélection des SSB_{direct}, des simulations similaires à celles décrites en section 4.1.1 sont menées. Ainsi, la géométrie utilisée est un noyau de fibroblaste contenant un génome humain mâle et les particules primaires utilisées sont des protons dont les énergies cinétiques vont de 0,5 à 20 MeV (TEL de 2,8 à

42,6 keV/ μm). Cependant, ces simulations ont pour particularité de prendre en compte différents critères de sélection des $\text{SSB}_{\text{direct}}$:

1. Tout dépôt d'énergie généré par une excitation ou ionisation et localisé dans le squelette d'un nucléotide ("backbone") conduit à la création d'un $\text{SSB}_{\text{direct}}$, à condition toutefois qu'il n'en existe pas déjà un au même endroit. Cette condition est très maximisante du point de vue de la génération des $\text{SSB}_{\text{direct}}$ car les nombreuses excitations qui déposent de faibles quantités d'énergie (<10 eV) sont toutes capables d'occasionner un $\text{SSB}_{\text{direct}}$.
2. Le seuil appliqué à l'énergie déposée dans le squelette d'un nucléotide est successivement fixé à 12,5 ; 17,5 et 30 eV.
3. Le seuil appliqué à l'énergie déposée est remplacée par une probabilité d'acceptation variant linéairement entre 5 et 37,5 eV [FRIEDLAND et al., 2003]. Plus précisément, cette probabilité est caractérisée par : $P(E_{\text{dep}} \leq 5 \text{ eV}) = 0$ et $P(E_{\text{dep}} \geq 37,5 \text{ eV}) = 1$.

Lors de chaque simulation, le nombre moyen de DSB/event calculé est converti en un nombre moyen de DSB/Gy/Gbp comme indiqué en section 4.1.1.

Durée simulée de l'étape chimique

Pour estimer l'influence de la durée de l'étape chimique, des simulations spécifiques ont été menées à l'échelle de quelques voxels. Quatre voxels "avant" (cf. section 2.2.2) sont ainsi disposés consécutivement de manière à former un filament de chromatine rectiligne. Notons que ce filament est similaire à la fibre de chromatine "compactée" introduite en figure 2.17a (section 2.2.2) mais découpée en quatre voxels. L'ADN contenu dans les voxels est constitué d'eau liquide uniquement. Les quatre voxels ainsi disposés sont irradiés avec des particules primaires envoyées dans leur direction depuis la surface d'un cylindre les englobant. Ce cylindre est caractérisé par un rayon de 200 nm et une longueur de 400 nm. Les particules primaires utilisées sont :

- Protons de 0,5 MeV avec un TEL de 42,6 keV/ μm .
- Protons de 9,62 MeV avec un TEL de 5 keV/ μm .
- Ions alpha de 17,39 MeV avec un TEL de 36 keV/ μm .

Les paramètres associés par défaut aux simulations sont ceux présents par défauts à l'exception de la durée de l'étape chimique et du critère de sélection des $\text{SSB}_{\text{direct}}$. Plusieurs durées de l'étape chimique sont testées pour chaque type de particule primaires simulé. Ces durées sont réparties de 1 ns à 2 μs et 10000 particules primaires sont simulées pour chacune d'entre elles. Dans un premier temps, l'ensemble des simulations est effectué avec le critère de sélection présent par défaut dans FullSim, soit le seuil en énergie de 17,5 eV. Ensuite, ce critère est modifié pour correspondre à la probabilité d'acceptation variant linéairement entre 5 et 37,5 eV. Enfin, les résultats produits par chaque simulation sont analysés de manière à obtenir le nombre moyen de DSB simulés par particule primaire.

Influence des voxels isolés

Afin d'estimer l'influence du découpage en voxels isolés de la chromatine, des simulations ont été menées dans deux géométries correspondant respectivement à une succession de 4 voxels "avant" (cf. section 2.2.2) et la fibre d'hétérochromatine introduite en figure 2.17a et section 2.2.2. Ces deux géométries sont illustrées en figure 4.14. Pour

chaque type de géométrie, des particules primaires identiques ont été simulées de manière à couvrir un large spectre de TEL :

- Protons de 9,62 MeV (TEL de 5 keV/μm).
- Alpha de 1,66 MeV (TEL de 160 keV/μm).
- Alpha de 5,72 MeV (TEL de 90 keV/μm).
- Alpha de 17,39 MeV (TEL de 37 keV/μm).

En effet, les 4 voxels consécutifs constituent une "fibre" moins compacte que la fibre d'hétérochromatine en raison de l'espace inoccupé situé à leurs extrémités. Ainsi, bien que la fibre de chromatine constituée par les voxels "avant" soit continue, la structure interne des voxels ne permet pas d'assurer un degré de compaction identique à celui de la fibre d'hétérochromatine. Ceci est illustré en figure 4.14b et dans le tableau 4.3. Le tableau 4.3 indique le nombre de paires de nucléotides (ou paires de bases "bp") ainsi que le nombre de paires de nucléotides par unité de longueur ("bp/μm") pour chacune des deux géométries considérées. De plus, l'utilisation des voxels implique que la simulation de l'étape chimique est effectuée indépendamment pour chaque couple event/voxel tel que mentionné en section 3.2. Cette "indépendance" isole les voxels les uns des autres en empêchant tout transfert d'espèce chimique ou toute réaction entre deux voxels distincts.

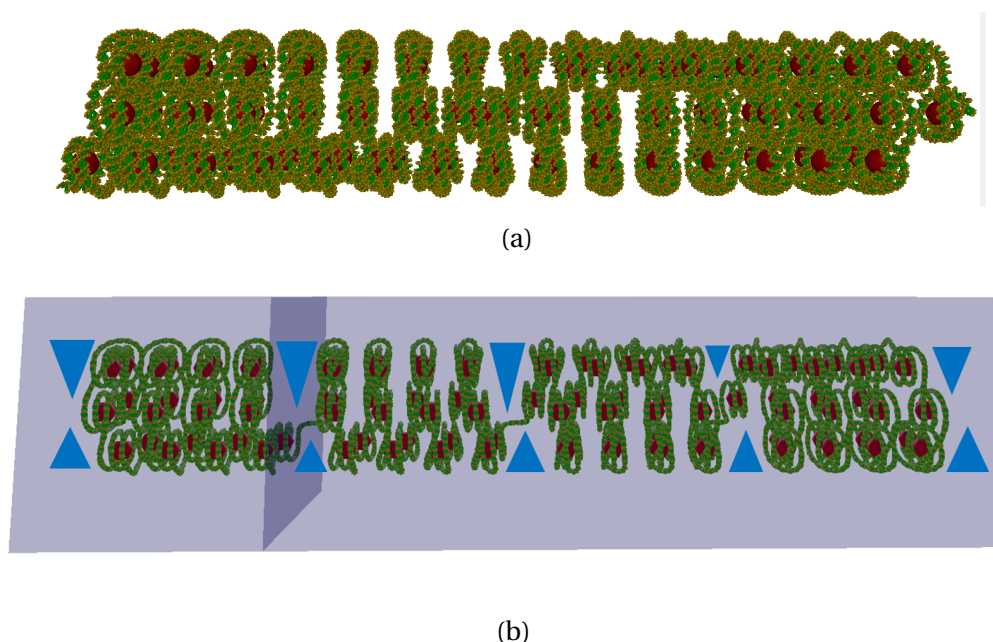


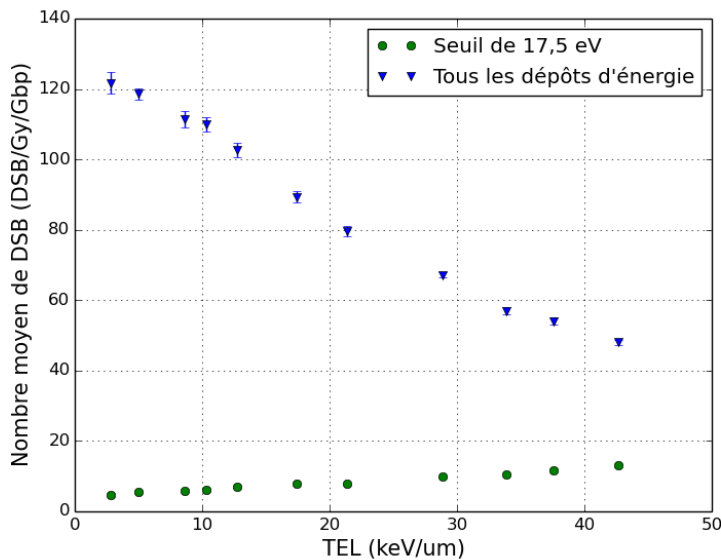
FIGURE 4.14 – Deux fibres de chromatine construites avec DnaFabric : une fibre d'hétérochromatine (a) et une fibre constituée de 4 voxels "avant" consécutifs (b). Les triangles bleus présents sur en (b) indiquent des zones de "vide". Ces zones sont de taille et de forme constante mais les surfaces des triangles les représentant varient en raison d'une distorsion liée à la projection en 2D de la fibre.

TABEAU 4.3 – Nombre total de paires de nucléotides ("bp") et par unité de longueur ("bp/μm") pour une fibre d'hétérochromatine et une fibre constituée de 4 voxels "avant" consécutifs.

Géométrie	bp	bp/μm
Fibre d'hétérochromatine	18000	113
4 voxels "avant"	19440	97.2

4.3.2 Résultats : sélection des dommages directs

La figure 4.15 décrit l'évolution du nombre moyen de DSB/Gy/Gbp simulés en fonction du TEL des protons incidents et de la présence d'un critère de sélection pour l'identification des SSB_{direct} . L'absence d'un tel critère de sélection signifie que tous les dépôts d'énergie localisés dans le squelette d'un nucléotide sont capables de générer un SSB_{direct} . La figure 4.15 montre que cela induit un nombre moyen de DSB/Gy/Gbp qui diminue avec l'énergie des protons primaires utilisés. En effet, le nombre de DSB/Gy/Gbp débute à 122 pour 2,8 keV/ μ m et atteint 49 pour 46,2 keV/ μ m, soit une décroissance de 60%. Lorsque le critère de sélection est celui appliqué par défaut (seuil de 17,5 eV) alors le nombre moyen de DSB/Gy/Gbp simulés est celui précédemment introduit en section 4.1.2 et figure 4.3. Précisons que, comme indiqué en section 4.1.2, ce nombre augmente légèrement en fonction du TEL du proton incident pour atteindre un maximum de 13 DSB/Gy/Gbp avec un TEL de 42,6 keV/ μ m. Cependant et malgré cette augmentation, le nombre de DSB/Gy/Gbp simulés quelque soit le dépôt d'énergie est systématiquement et largement plus important que celui obtenu avec le critère de sélection par défaut. Remarquons que l'écart entre les deux résultats décroît avec l'augmentation du TEL des protons primaires utilisés. En effet, les DSB/Gy/Gbp obtenus sans critère de sélection sont supérieurs d'un facteur 23 à ceux calculés avec le critère par défaut pour un TEL 2,8 keV/ μ m tandis que pour un TEL de 42,6 keV/ μ m cet écart est de 276%. Il est intéressant de noter que l'absence de critère de sélection des SSB_{direct} change radicalement l'évolution du nombre de DSB/Gy/Gbp simulés : ce dernier ne croît plus avec le TEL mais, au contraire, diminue.



(a)

FIGURE 4.15 – Influence de la présence d'un critère en énergie déposée dans le squelette d'un nucléotide pour la sélection des SSB_{direct} sur le nombre moyen de DSB/Gy/Gbp simulés. Les nombre moyens de DSB, calculés en comptabilisant tous les dépôts d'énergie en tant que SSB_{direct} , sont représentés par des points bleus tandis ceux obtenus avec le critère de sélection par défaut dans FullSim (seuil de 17,5 eV) sont illustrés par des points verts.

Les effets provoqués par l'application d'un critère de sélection constitué d'une probabilité d'acceptation ($P(5eV) = 0$ et $P(37,5eV) = 1$) [FRIEDLAND et al., 2003] sont illus-

trés en figure 4.16. L'application de cette probabilité d'acceptation augmente le nombre moyen de DSB/Gy/Gbp calculés de manière constante pour tous les TEL considérés. Il s'agit d'une augmentation de 3 DSB/Gy/Gbp en moyenne. Comme cette augmentation est constante, l'évolution du nombre moyen de DSB/Gy/Gbp augmente de la même manière avec le TEL.

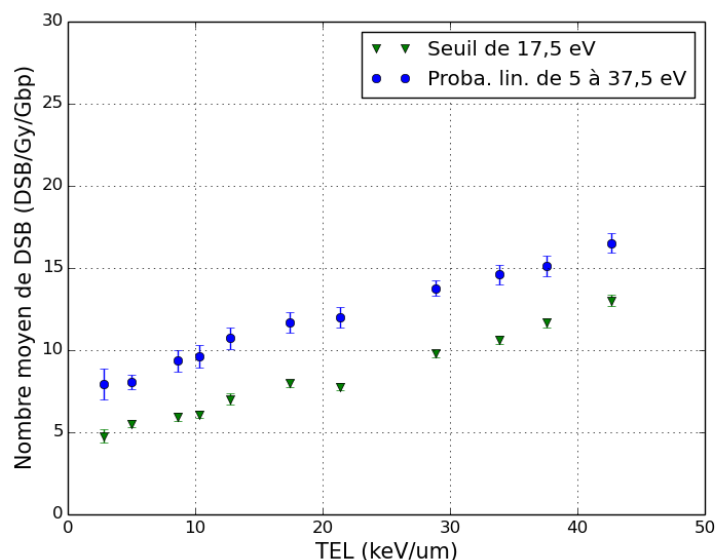


FIGURE 4.16 – Effet sur le nombre moyen de DSB/Gy/Gbp calculé à partir de deux critères de sélection concernant les SSB_{direct} : le critère sélection par défaut (seuil de 17,5 eV) et la probabilité d'acceptation variant linéairement avec l'énergie déposée dans le squelette d'un nucléotide.

La figure 4.17 présente l'évolution du nombre moyen de DSB/Gy/Gbp simulés en fonction du TEL des protons incidents et de trois critères de sélection des SSB_{direct} basés sur des seuils appliqués à l'énergie déposée dans le squelette d'un nucléotide : ces seuils sont de 12,5; 17,5 et 30 eV. Il apparaît que l'augmentation du nombre moyen de DSB/Gy/Gbp calculés avec le TEL est maintenue pour chacun des seuils testés. Remarquons également que l'application d'un seuil de 12,5 eV semble induire une augmentation avec le TEL plus prononcée pour les TEL inférieurs à 15 keV/ μ m. L'utilisation d'un seuil de 12,5 eV donne des résultats supérieurs à tous les autres pour tous les TEL étudiés. A l'inverse, l'usage d'un seuil de 30 eV entraîne des résultats systématiquement inférieurs. Les résultats obtenus par [NIKJOO et al., 2001] (seuil de 17,5 eV) sont également illustrés en figure 4.17 et sont particulièrement en bon accord avec le nombre moyen de DSB/Gy/Gbp simulés dans ce travail lorsqu'un seuil de 12,5 eV est appliqué.

4.3.3 Résultats : durée simulée de l'étape chimique

La figure 4.18 (cf. page 161) montre l'évolution du nombre moyen de DSB simulés avec FullSim en fonction de l'évolution de la durée de l'étape chimique et de deux critères de sélection des SSB_{direct} : le critère par défaut (seuil de 17,5 eV) et la probabilité d'acceptation linéaire entre 5 et 37,5 eV mentionnée précédemment. Les durées prises en compte pour l'étape chimique varient entre 1 et 2000 ns pour les trois types de particules primaires considérées : des protons de 0,5 MeV, de 10 MeV et des ions alpha de 17,39 MeV. La figure 4.18 montre que quelque soit le critère ou la particule primaire utilisée, le nombre de DSB/event simulés est toujours minimal lorsque la durée de l'étape chimique est de 1

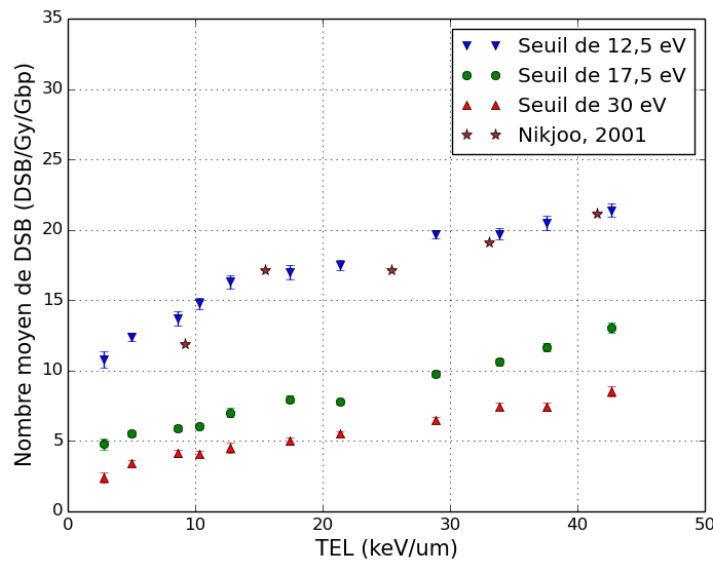


FIGURE 4.17 – Application de trois critères de sélection différents pour l'identification des SSB_{direct} lors du calcul du nombre moyen de DSB/Gy/Gbp. Les critères de sélection utilisés sont des seuils en énergie valant respectivement 12,5 (triangles inversés bleus) ; 17,5 (points verts) et 30 eV (triangles rouges). Les résultats de [Nikjoo et al., 2001] sont également présents (étoiles marrons).

ns. Il apparaît également que ce minimum est plus élevé lorsque la probabilité linéaire est utilisée par rapport à celui obtenu avec le seuil de 17,5 eV. Remarquons également que, pour toutes les durées simulées, le nombre de DSB moyen calculé par particule primaire est le plus faible dans le cas des protons de 10 MeV qui ont le TEL le plus faible et le plus important pour les protons de 0,5 MeV qui présentent le TEL le plus élevé. Lorsque le critère de sélection des SSB_{direct} correspond au seuil de 17,5 eV alors le nombre moyen de DSB simulé augmente avec la durée de l'étape chimique. Dans l'ensemble, le nombre moyen de DSB simulé atteint une valeur stable lorsque la durée de l'étape chimique se situe entre 10 et 100 ns et ce pour les trois particules primaires utilisées. Remarquons que les valeurs sont particulièrement dispersées dans le cas du proton de 10 MeV et des durées supérieures à 100 ns. Cette dispersion provient des incertitudes statistiques qui sont particulièrement élevées dans ce cas précis. Le tableau 4.4 présente les écarts relatifs entre le nombre moyen de DSB obtenu avec une étape chimique de 2,5 ns (valeur par défaut de FullSim) et celui calculé avec une étape chimique dont la durée est suffisante pour atteindre la zone de stabilité. Ces écarts vont de +34,6% dans le cas des ions alpha de 17,39 MeV à +26% avec des protons de 0,5 MeV. Les résultats simulés avec la probabilité d'acceptation linéaire comme critère de sélection des SSB_{direct} (cf. figure 4.18) indiquent une augmentation du nombre moyen de DSB simulé avec la durée de l'étape chimique. De façon similaire à ce qui a été précédemment observé, le nombre moyen de DSB atteint une valeur stable à 10 ns pour les protons 0,5 MeV, 10 MeV ainsi que pour les alpha de 17,39 MeV.

4.3.4 Résultats : influence des voxels isolés

La figure 4.19 (cf. page 162) illustre les nombres moyen de DSB calculés avec la configuration par défaut dans une portion de chromatine. Cette portion de chromatine est d'abord représentée par une unique fibre d'ADN puis par 4 voxels "avants" consécu-

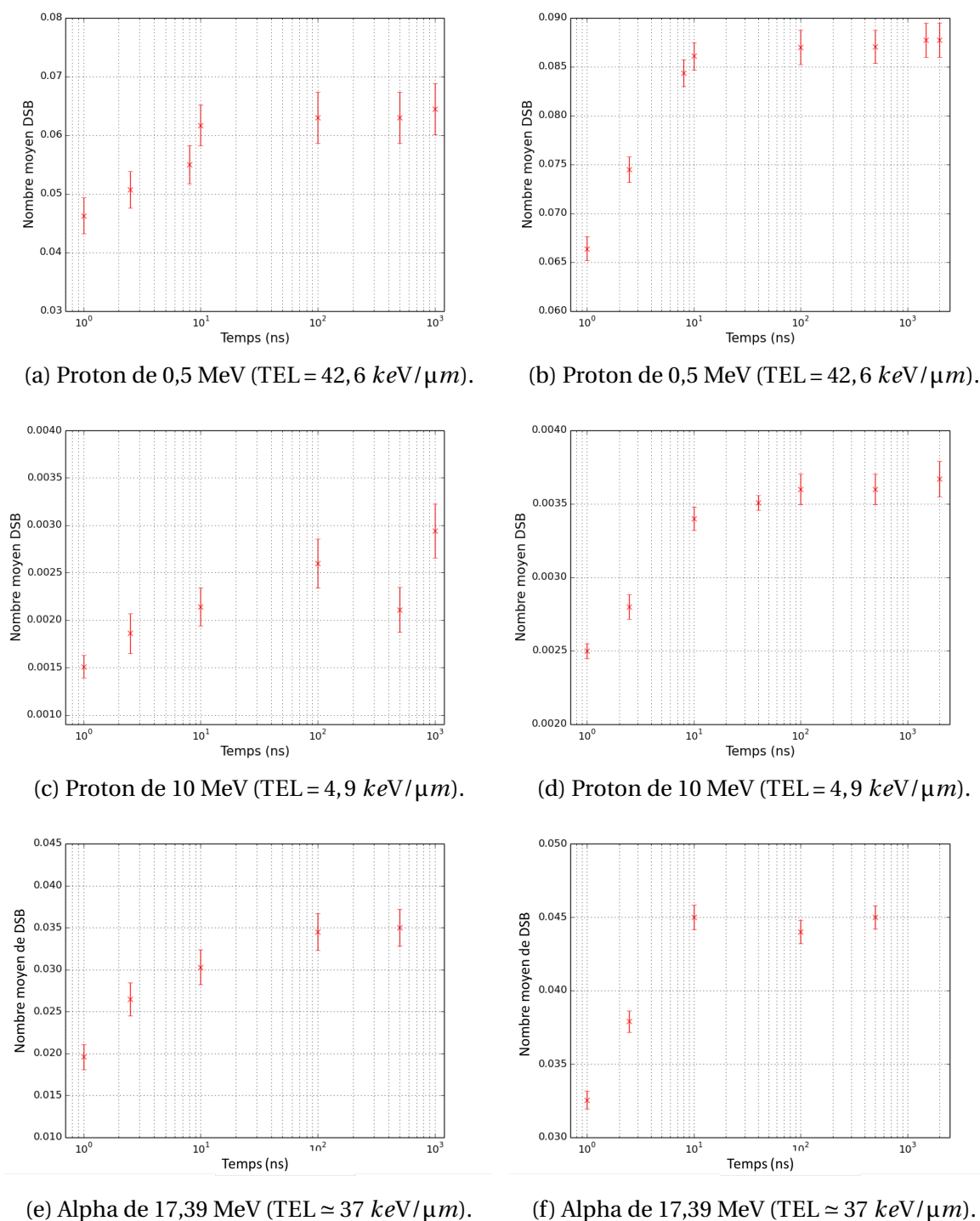


FIGURE 4.18 – Le nombre moyen de DSB par particule primaire calculé avec FullSim en fonction de différentes durées de l'étape chimique. Ces résultats proviennent de la simulation de l'irradiation d'un noyau de type fibroblastique et deux critères de sélection des SSB_{direct} sont successivement utilisés : le seuil de 17,5 eV [(a), (c) et (e)] et la probabilité d'acceptation linéaire entre 5 et 37,5 eV [(b), (d) et (f)]. Les particules primaires simulées sont des protons de 0.5 MeV (TEL de 42,6 keV/μm) [(a), (b)] et 10 MeV (TEL de 5 keV/μm) [(c), (d)] ainsi que des ions alpha de 17,39 MeV (TEL ≈ 37 keV/μm) [(e), (f)].

TABLEAU 4.4 – Augmentation du nombre de DSB moyen calculés avec FullSim lorsque la durée de l'étape chimique varie de 2,5 ns au seuil de stabilisation du nombre moyen de DSB. Cette augmentation ("Diff") est calculée à partir de la formule suivante : $(DSB_{max} - DSB_{FullSim}) / DSB_{FullSim}$.

Particule primaire	TEL (keV/ μm)	$DSB_{FullSim}$ (2,5 ns)	DSB_{max}	Diff (%)
Proton 0,5 MeV	42,6	$5 \cdot 10^{-2}$	$6,3 \cdot 10^{-2}$	26
Alpha 17,39 MeV	37,3	$2,6 \cdot 10^{-2}$	$3,5 \cdot 10^{-2}$	34,6
Proton 10 MeV	4,9	$1,9 \cdot 10^{-3}$	$2,5 \cdot 10^{-3}$	31,6

tifs. Les résultats obtenus montrent que le fait d'effectuer une simulation dans 4 voxels "avants" ou dans une fibre conduit à des nombres moyens de DSB par particule primaire similaires. En effet, l'écart maximal est obtenu pour les ions alpha de 1,66 MeV avec un nombre moyen de DSB/event enregistré dans les voxels qui est supérieur de 12% à celui calculé dans la fibre. Concernant les protons de 9,62 MeV, les nombres moyens de DSB/event simulés sont identiques quelque soit la géométrie utilisée (voxels ou fibre). Au delà du fait que les résultats obtenus soient très similaires avec une géométrie constituée de voxels ou d'une fibre, il apparaît que les nombres moyens de DSB simulés avec des ions alpha sont légèrement inférieurs lorsque la fibre est utilisée. Insistons sur le fait que cette constatation est valable pour les trois ions alpha qui présentent de forts TEL (TEL de 160, 90 et 36 keV/ μm) mais pas pour les protons de 9,62 MeV de plus faible TEL (TEL de 5 keV/ μm).

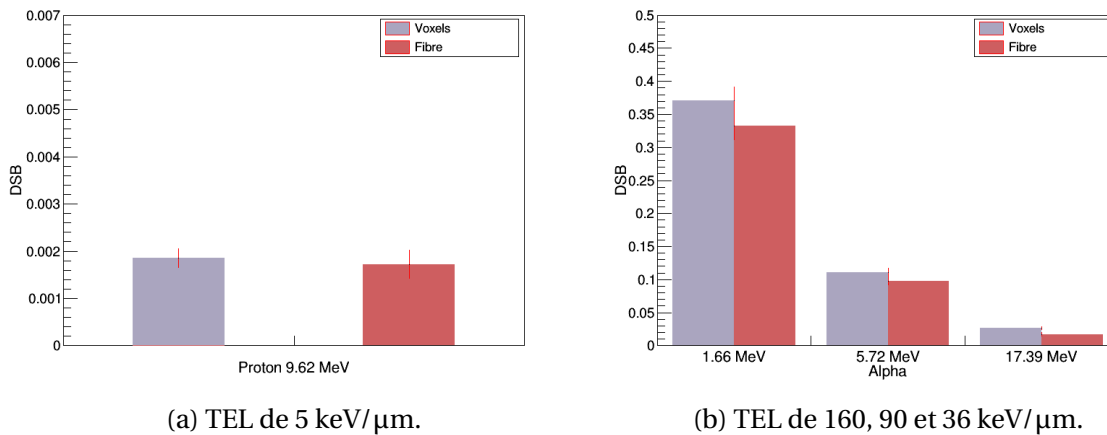


FIGURE 4.19 – Le nombre de DSB calculés par particule primaire avec FullSim dans deux géométries distinctes : une suite de 4 voxels "avant" consécutifs (gris-bleu) et une fibre continue de 90 nucléosomes (rouge). Les particules primaires utilisées sont des protons de 9,62 MeV (a) et des ions alpha de 1,66 ; 5,72 et 17,39 MeV (b).

4.3.5 Discussion

Importance de la sélection des dommages à l'ADN issus de l'étape physique pour l'équilibre de la simulation

L'utilisation d'un critère de sélection plus ou moins contraignant pour identifier les SSB_{direct} s'avère avoir une influence très significative sur le nombre moyen de DSB/-Gy/Gbp calculés. En effet, la figure 4.15 montre que le fait de considérer que tous les dépôts d'énergie peuvent donner lieu à des SSB_{direct} conduit à l'obtention d'un nombre de

DSB/Gy/Gbp systématiquement élevé. Ces quantités particulièrement importantes proviennent du fait que tout dépôt d'énergie, induit par une ionisation, une excitation ou encore une excitation vibrationnelle, ayant lieu dans l'ADN conduit à la création d'un SSB_{direct} . En effet, ces dépôts sont particulièrement nombreux en raison, notamment, de la prise en compte des excitations vibrationnelles qui sont particulièrement fréquentes et déposent de faibles quantités d'énergie par interaction ($< eV$). En raison de ces faibles quantités d'énergie déposées, il est communément admis que ces interactions ont une très faible probabilité de générer des SSB_{direct} et c'est pourquoi un critère de sélection est présent dans notre chaîne de calcul. En tenant compte de tous les dépôts d'énergie, les nombreux SSB_{direct} ainsi générés ont une probabilité significative d'être distants de moins de 10 paires de nucléotides tout en étant situés sur des brins d'ADN différents. Et, ainsi, ils augmentent considérablement le nombre de DSB produits par la simulation. La présence d'une sélection, basée sur l'énergie déposée dans un nucléotide, élimine la prise en compte de la majorité des excitations vibrationnelles car elles ne déposent que peu d'énergie lors de chaque interaction. Ainsi, la mise en place d'une sélection des dépôts d'énergie dans le squelette de l'ADN pouvant conduire à un SSB_{direct} , diminue significativement le nombre de DSB/Gy/Gbp simulé. La figure 4.16 montre que l'utilisation d'une probabilité d'acceptation linéaire constitue un critère moins restrictif que l'usage d'un simple seuil de 17,5 eV signifiant ainsi qu'une plus grande proportion des faibles dépôts d'énergie est capable de franchir ce filtre.

Lorsque tous les dépôts d'énergie peuvent générer des SSB_{direct} (absence de sélection), une diminution du nombre moyen de DSB/Gy/Gbp simulés avec l'augmentation du TEL des protons primaires peut être observée en figure 4.15. Et, à l'inverse, lorsqu'un critère de sélection est utilisé alors les SSB_{direct} augmentent avec le TEL. Ces variations sont liées au fait que les résultats soient tous présentés par Gray (DSB/Gy/Gbp) déposé dans le noyau. En effet, lorsque le TEL des protons primaires augmente, il devient nécessaire de diminuer le nombre de protons primaires pris en compte. Cette diminution du nombre de particules primaires a pour effet de faire diminuer mécaniquement la quantité de DSB/Gy/event car moins de particules sont simulées. Et l'importance de cette diminution s'avère d'autant plus significative que le nombre de DSB/event est élevé. Pour comprendre cela, il est nécessaire de considérer à nouveau l'équation 4.1 introduite en section 4.1.1 et rappelée ci-dessous :

$$N_{DSB/Gy/Gbp}(s_{bp}) = N_{DSB/event}(s_{bp}) \cdot \frac{E_{1Gy}}{\bar{l} \cdot TEL(E_p) \cdot n}$$

Cette équation permet de calculer le nombre moyen de DSB/Gy/Gbp ($N_{DSB/Gy/Gbp}(s_{bp})$) à partir du nombre de DSB/event ($N_{DSB/event}(s_{bp})$) directement généré par la simulation et du TEL ($TEL(E_p)$) de la particule primaire utilisée. Seuls deux éléments de cette équation varient dans les simulations : le nombre de DSB/event générés et le TEL. Les influences de ces deux éléments s'opposent et l'évolution du nombre de DSB/Gy/Gbp résulte de leur combinaison. Dans les simulations réalisées, des particules primaires sont simulées avec des TEL de plus en plus élevés. En considérant les TEL minimum et maximum utilisés dans nos simulation, le TEL augmente d'un facteur 15. En parallèle, le nombre de DSB/event générés augmente également au fur et à mesure que le TEL croît. Cependant, l'augmentation du nombre de DSB/event n'est pas constante et dépend du critère de sélection des SSB_{direct} retenu. En effet, le nombre de DSB/event entre les TEL minimal et maximal augmente d'un facteur 6 en l'absence de critère de sélection et d'un facteur 41 lorsque le critère de sélection retenu est le seuil de 17,5 eV. Ces augmentations relatives montrent bien qu'en l'absence de critère de sélection des SSB_{direct} ce sont les variations

du TEL qui sont les plus importantes (facteur 15 contre 6) ce qui se traduit par une diminution du nombre de DSB/Gy/Gbp calculés (cf. équation 4.1). Inversement, lorsque le critère de sélection retenu est le seuil de 17,5 eV, ce sont les variations du nombre de DSB/event qui sont dominantes (facteur 41 contre 15) et cela implique que le nombre de DSB/Gy/event calculés augmente (cf. équation 4.1). Les résultats présentés en figure 4.17 vont également dans le sens de cette explication car l'usage d'un critère de sélection moins restrictif adoucit l'augmentation du nombre de DSB/Gy/event avec le TEL.

Finalement, les variations induites par les divers critères de sélections mis en œuvre en figure 4.15, 4.17 et 4.16 démontrent l'influence majeure de ce paramètre. Une telle sélection semble donc nécessaire et une explication peut être avancée : le critère de sélection correspond à un processus physique réel et non considéré en tant que tel dans la simulation. L'application du critère de sélection revient alors à simuler indirectement ce processus en attribuant à chaque nucléotide, dans lequel un ou plusieurs dépôts d'énergie sont localisés, une probabilité de devenir un SSB_{direct} . Toutefois, la prise en compte d'un tel processus physique dépasse le cadre de ce travail. En effet, il s'agirait d'étudier finement la structure du nucléotide et son environnement, d'un point de vue quantique par exemple, pour pouvoir accéder au détail des effets occasionnés par une interaction sur un nucléotide.

Remarquons que l'usage d'un critère constitué d'un seuil de 12,5 eV permet à notre simulation de reproduire les résultats simulés de [NIKJOO et al., 2001] (cf. figure 4.17). En fait, l'usage d'un critère de sélection influe sur l'équilibre entre le nombre de SSB_{direct} obtenu et le nombre de traces requises pour déposer 1 Gy dans le noyau cellulaire. Et, sans cet équilibre, l'évolution du nombre de DSB/Gy/Gbp simulés en fonction de l'énergie des protons primaires (cf. figure 4.15) ne correspond pas à celle expérimentalement observée (cf. figure 4.3 en section 4.1.2).

Prise en compte du "scavenging" des radicaux dans la simulation

La limitation de la durée de l'étape chimique est utilisée dans notre simulation comme un moyen de prendre en compte le "scavenging" des radicaux OH^* dans le noyau cellulaire. Et c'est pour cette raison qu'elle est limitée par défaut à 2,5 ns tandis que celle simulée dans le code PARTRAC est de 10 ns par exemple. A ce propos, la figure 4.18 montre que la présence d'une telle limitation a un effet relativement constant quelque soit le TEL et la particule primaire utilisée lorsque FullSim est dans sa configuration par défaut. En effet, la figure 4.19 indique que le fait de laisser la simulation de l'étape chimique continuer jusqu'à l'équilibre (en terme de DSB/Gy/Gbp simulé) entraîne des augmentations du nombre de DSB/Gy/Gbp simulé autour de 30% lorsque le critère de sélection des SSB_{direct} est le seuil de 17,5 eV. Cependant, si le critère de sélection des SSB_{direct} est la probabilité d'acceptation linéaire alors le fait de laisser la simulation de l'étape chimique continuer jusqu'à l'équilibre entraîne une augmentation plus importante du nombre moyen de DSB simulé dans le cas des protons de 10 MeV (TEL de 5 keV/ μ m). Ceci signifie que la prise en compte du scavenging des radicaux OH^* peut avoir des effets différents en fonction des caractéristiques des particules simulées si le critère de sélection est la probabilité d'acceptation variant linéairement. Les résultats générés avec la configuration par défaut de FullSim présentent une précision statistique moins bonne que celle obtenue avec une probabilité d'acceptation linéaire. Cela provient du fait que le critère de sélection des SSB_{direct} de la configuration par défaut est plus restrictif que la probabilité linéaire. En effet, moins de SSB_{direct} et de DSB sont simulés avec la configuration par défaut et cela dégrade la précision statistique des résultats à temps de calcul égal.

Finalement, la prise en compte des "scavengers" actuellement réalisée dans notre simulation permet de générer des résultats acceptables vis-à-vis des données expérimentales (cf. la figure 4.3 en section 4.1.2) avec des protons mais les comparaisons n'ont été effectuées que dans le cadre de protons et pas avec des ions alpha. Au delà de ce résultats, remarquons que les processus biologiques et chimiques impliqués dans le scavenging des espèces chimiques produites lors de la radiolyse de l'eau se produisent de manière continue dans le temps. Aussi, une prise en compte du "scavenging" de manière continue tout au long de la simulation de l'étape chimique semble plus réaliste. Cela aurait cependant pour effet d'allonger le temps de simulation et c'est la raison pour laquelle cela n'a pas été effectué dans ce travail de thèse.

Inadéquation du TEL pour la caractérisation des particules primaires

En section 1.1.1, le TEL a été introduit en tant que grandeur ne permettant pas de caractériser totalement la répartition des dépôts d'énergie induits par une particule primaire donnée. Les résultats de la figure 4.18 illustrent cela en mettant en évidence le fait que deux particules ayant des TEL proches peuvent générer un nombre moyen de DSB très différent. En effet, les ions alpha ont un TEL de 37,3 keV/ μm et les protons de 0,5 MeV ont un TEL de 42,6 keV/ μm , soit un écart relatif de +14% qui peut être considérée comme faible dans ce contexte, et génèrent pourtant des nombres moyens de DSB très différents. Ainsi, pour une limitation à 2,5 ns de la durée de l'étape chimique, les ions alpha de 17,39 MeV génèrent 0,026 DSB contre 0,05 DSB pour les protons de 0,5 MeV. Ceci équivaut à une différence relative de +92% au niveau du nombre moyen de DSB simulés ce qui est bien plus élevé que les +14% d'écart constatés à propos du TEL. L'inadéquation du TEL provient du fait qu'il ne permet pas de rendre compte de l'énergie cinétique associée aux particules secondaires. En effet, les protons de 0,5 MeV et les ions alpha de 17,39 MeV ont des TEL similaires mais des vitesses bien différentes ce qui se traduit par des particules secondaires émises avec différentes énergies cinétiques. Ainsi, les particules secondaires émises par les ions alpha de 17,39 MeV ont des énergies cinétiques bien plus importantes que celles générées par les protons de 0,5 MeV. En conséquence, les dépôts d'énergie induits par les protons de 0,5 MeV sont plus concentrés et les dommages directs causés à l'ADN plus nombreux [DOS SANTOS, 2013]. A ce propos, les résultats de la figure 4.18 montrent que la prise en compte de l'étape chimique ne change pas cet état de fait.

Impact du découpage de la chromatine en voxels isolés lors de la simulation de l'étape chimique

Rappelons que l'utilisation dans ce travail d'une modélisation géométrique de la chromatine basée sur des voxels permet d'optimiser l'usage de la mémoire ce qui rend possible l'insertion d'un génome humain complet dans les noyaux cellulaires simulés (cf. chapitre 2). Les voxels sont également nécessaires pour la simulation de l'étape chimique qu'ils permettent de subdiviser en plusieurs simulations intermédiaires basées sur la notion de couple event/voxel (cf. section 3.2). D'un point de vue pratique, l'utilisation de voxels implique toutefois que :

- Comme indiqué en section 4.3.1, l'utilisation des voxels entraîne l'apparition d'espaces "vides" à leurs frontières. Ces espaces "vides" diminuent légèrement le nombre de nucléotides présents par unité de longueur.
- Le découpage de la chromatine en voxels est lié à la simulation de l'étape chimique qui est effectuée voxel par voxel. En effet, les voxels ainsi pris en considération sont

indépendants et isolés les uns des autres et cela signifie que les espèces chimiques ne peuvent pas diffuser en dehors de leur voxel d'origine. Par conséquent, les espèces chimiques simulées ne peuvent réagir avec le contenu des autres voxels limitant ainsi leur capacité à endommager l'ADN et à réagir entre elles.

Les résultats présentés en figure 4.19 montrent que l'influence de ces deux éléments est faible dans le cas du calcul d'un nombre moyen de DSB, à fortiori lorsque des protons sont utilisés en tant que particules primaires. Ceci est une constatation importante car la chaîne de calcul est, en bonne partie, basée sur cette approximation dont l'impact s'avère finalement limité.

Toutefois, il est intéressant de remarquer que le fait de découper la chromatine en voxels a des effets différents selon la particule primaire utilisée et son TEL. A ce propos, la figure 4.19 montre que l'utilisation d'une fibre ou d'un voxel avec des protons de 9,62 MeV génère des résultats très similaires en termes de DSB/event. Avec des ions alpha de 1,66 ; 5,72 et 17,39 MeV, l'utilisation de voxels entraîne un nombre de DSB calculé plus important qu'avec la fibre. L'origine de ces différents comportements provient des deux effets mentionnés précédemment, c'est à dire les "espaces vides" et l'isolement des voxels lors de l'étape chimique. D'un point de vue théorique, si le premier effet (espaces "vides") avait une influence majoritaire alors c'est dans la fibre que le nombre de DSB calculé est le plus important car la fibre a le plus grand nombre de paires de nucléotides par unité de longueur et possède donc un ADN plus concentré que celui des voxels. Cette plus grande concentration facilite l'apparition de DSB en rendant plus probable la création des clusters de dommages à l'ADN. Toutefois, cela ne correspond pas aux résultats de la figure 4.19 où le nombre de DSB calculé n'est pas plus important lorsque la fibre est utilisée. Si c'était le second effet qui a une influence majoritaire alors ce sont dans les voxels que la quantité de DSB simulés sera la plus importante. Ceci provient du fait que l'utilisation d'une fibre permet aux espèces chimiques de réagir "librement" entre elles ce qui n'est pas le cas avec une succession de voxels isolés. En effet, les espèces chimiques sont alors confinées dans leur voxel d'origine et, entre autres, le radical OH^\bullet a une probabilité plus faible de rencontrer une espèce chimique capable de le neutraliser (cf. tableau 1.5, section 1.3.2). Cette probabilité plus faible implique que le OH^\bullet peut rester, en moyenne, plus longtemps dans la simulation et, donc, qu'il présente une probabilité plus importante d'endommager l'ADN ($\text{SSB}_{\text{indirect}}$). Ceci explique pourquoi l'usage de voxels augmenterait le nombre de DSB/event simulés si l'influence du second effet était dominante. Les résultats de la figure 4.19 indiquent que le nombre de DSB simulés est légèrement plus important avec des voxels plutôt qu'avec la fibre lorsque les ions alpha sont utilisés (cf. figure 4.19b) et cela suggère que c'est l'influence du second effet qui est dominante.

Les explications avancées impliquent cependant que l'utilisation d'un critère de sélection des $\text{SSB}_{\text{direct}}$ pourrait rendre le premier effet (présence d'"espace vides") dominant. En effet, le choix d'un critère de sélection moins restrictif, tel que la probabilité d'acceptation linéaire par exemple, entraînerait un plus grand nombre de $\text{SSB}_{\text{direct}}$ issus de l'étape physique et cela augmenterait l'importance associée au nombre de nucléotides par unité de longueur. Et comme le nombre de nucléotides par unité de longueur dépend directement de la présence des "espaces vides", le premier effet prendrait alors de l'importance. Ceci pourrait alors rendre majoritaire le nombre de DSB simulés lorsque la géométrie utilisée n'est pas découpée en voxels.

Références

- BANTSAR, A. 2012, «Ionization cluster size distributions created by low energy electrons and alpha particles in nanometric track segment in gases», *arXiv preprint arXiv :1207.6893*. [146](#)
- BANTSAR, A., B. GROSSWENDT, S. PSZONA et J. KULA. 2009, «Single track nanodosimetry of low energy electrons», *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 599, n° 2-3, p. 270–274. [146](#)
- BELLI, M., R. CHERUBINI, M. DALLA VECCHIA, V. DINI, G. ESPOSITO, G. MOSCHINI, O. SAPORA, C. SIGNORETTI, G. SIMONE, E. SORRENTINO et M. TABOCCHINI. 2001, «DNA Fragmentation in mammalian cells exposed to various light ions», *Advances in Space Research*, vol. 27, n° 2, p. 393–399. [133](#), [136](#), [137](#), [138](#), [141](#), [142](#)
- BELLI, M., R. CHERUBINI, M. DALLA VECCHIA, V. DINI, G. MOSCHINI, C. SIGNORETTI, G. SIMONE, M. A. TABOCCHINI et P. TIVERON. 2000, «DNA DSB induction and rejoining in V79 cells irradiated with light ions: a constant field gel electrophoresis study», *International Journal of Radiation Biology*, vol. 76, n° 8, p. 1095–1104. [133](#), [136](#), [137](#), [138](#), [141](#), [142](#), [145](#)
- BERNAL, M. A., C. E. DEALMEIDA, S. INCERTI, C. CHAMPION, V. IVANCHENKO et Z. FRANCIS. 2015, «The Influence of DNA Configuration on the Direct Strand Break Yield», *Computational and Mathematical Methods in Medicine*, vol. 2015, p. 1–8. [133](#)
- BUG, M. U. 2014, *Nanodosimetric particle track simulations in water and DNA media*, thèse de doctorat, University of Wollongong, School of Physics. [154](#)
- BUG, M. U., E. SURDUTOVICH, H. RABUS, A. B. ROSENFELD et A. V. SOLOV'YOV. 2012, «Nanoscale characterization of ion tracks : MC simulations versus analytical approach», . [154](#)
- CAMPA, A., F. BALLARINI, M. BELLI, R. CHERUBINI, V. DINI, G. ESPOSITO, W. FRIEDLAND, S. GERARDI, S. MOLINELLI, A. OTTOLENGHI, H. PARETZKE, G. SIMONE et M. A. TABOCCHINI. 2005, «DNA DSB induced in human cells by charged particles and gamma rays: Experimental results and theoretical approaches», *International Journal of Radiation Biology*, vol. 81, n° 11, p. 841–854. [133](#), [136](#), [137](#), [138](#), [142](#)
- DOS SANTOS, M. 2013, *Modélisation de la topologie des dépôts d'énergie créés par un rayonnement ionisant à l'échelle nanométrique dans les noyaux cellulaires et relation avec les événements précoces radio-induits*, thèse de doctorat, Bordeaux I, Ecole doctorale des sciences physiques et de l'ingénieur. [143](#), [165](#)
- FRANKENBERG, D., H. J. BREDE, U. J. SCHREWE, C. STEINMETZ, M. FRANKENBERG-SCHWAGER, G. KASTEN et E. PRALLE. 1999, «Induction of DNA Double-Strand Breaks by 1h and 4he Ions in Primary Human Skin Fibroblasts in the LET Range of 8 to 124 keV/um», *Radiation Research*, vol. 151, p. 540–549. [133](#), [136](#), [137](#), [138](#), [141](#), [142](#), [144](#)
- FRIEDLAND, W., P. JACOB, P. BERNHARDT, H. G. PARETZKE et M. DINGFELDER. 2003, «Simulation of DNA damage after proton irradiation», *Radiation Research*, vol. 159, p. 401–410. [133](#), [134](#), [135](#), [137](#), [138](#), [139](#), [143](#), [145](#), [146](#), [156](#), [158](#)

- FRIEDLAND, W., P. JACOB et P. KUNDRAT. 2010, «Mechanistic simulation of radiation damage to DNA and its repair : on the track towards systems radiation biology modelling», *Radiation Protection Dosimetry*, vol. 143, n° 2-4, p. 542–548. [145](#)
- NIKJOO, H. et L. LINDBORG. 2010, «RBE of low energy electrons and photons», *Physics in Medicine and Biology*, vol. 55, n° 10, p. R65–R109. [145](#)
- NIKJOO, H., P. O'NEILL, W. E. WILSON et D. T. GOODHEAD. 2001, «Computational approach for determining the spectrum of DNA damage induced by ionizing radiation», . [133](#), [134](#), [138](#), [139](#), [143](#), [159](#), [160](#), [164](#)
- PATER, P., G. BÄCKSTÖM, F. VILLEGAS, A. AHNESJÖ, S. A. ENGER, J. SEUNTJENS et I. EL NAQA. 2016, «Proton and light ion RBE for the induction of direct DNA double strand breaks», *Medical Physics*, vol. 43, n° 5, p. 2131–2140. [145](#)
- PSZONA, S., J. KULA et S. MARJANSKA. 2000, «A new method for measuring ion clusters produced by charged particles in nanometre track sections of DNA size», *Nuclear Instruments and Methods in Physics Research Section A : Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 447, n° 3, p. 601–607. [146](#)
- SANTOS, M. D., C. VILLAGRASA, I. CLAIRAND et S. INCERTI. 2014, «Influence of the chromatin density on the number of direct clustered damages calculated for proton and alpha irradiations using a Monte Carlo code», *Progress in Nuclear Science and Technology*, vol. 4, p. 449–453. [133](#)

Conclusions et perspectives

Dans le cadre de l'amélioration de la compréhension du lien qui existe entre la caractérisation physique d'un rayonnement ionisant et son effet biologique, l'objectif de ce travail de thèse était le développement d'un outil capable de simuler les dommages radioinduits précoces à l'ADN générés suite à l'irradiation d'un noyau cellulaire par des ions légers. Un état de l'art de la compréhension actuelle de ce lien a été effectué dans le chapitre 1 et l'intérêt d'un tel outil dans notre projet de recherche explicité dans le cadre d'une approche mécanistique. Cette approche se caractérise, notamment, par la prise en compte détaillée de trois étapes successives dans la simulation conduisant aux effets biologiques précoces : les étapes physique, physico-chimique et chimique. Une autre caractéristique de cette approche est l'usage de représentations géométriques de l'ADN particulièrement détaillées. A ce propos, le chapitre 2 décrit le logiciel qui a été développé dans ce travail pour générer, manipuler et exporter de telles modélisations géométriques de l'ADN : DnaFabric [MEYLAN et al., 2016]. Avec DnaFabric, des modélisations géométriques ont été générés pour décrire à l'échelle moléculaire la totalité du génome humain, soit $6,4 \cdot 10^9$ paires de nucléotides contenues dans des noyaux de cellules humaines. Au delà de ces modélisations géométriques, la prise en compte des trois étapes précédemment mentionnées dans une simulation requiert des outils capables de répondre à des problématiques spécifiques : transport des particules, interactions physiques, dissociation des molécules d'eau, diffusion des espèces chimiques, réactions chimiques etc. De tels outils sont présents dans le code de simulation de Monte Carlo Geant4 et notamment dans son extension Geant4-DNA. La simulation construite dans ce travail de thèse est basée sur ce code de simulation mais des modifications spécifiques ont du être introduites pour répondre aux problématiques rencontrées dans ce travail. Il s'agissait notamment de permettre la prise en compte de la présence de l'ADN lors de la simulation de l'étape chimique. Les modifications effectuées dans Geant4 à cet effet seront prochainement mises à la disposition de la collaboration Geant4 et une application utilisateur pourrait être proposée pour en illustrer l'utilisation. La manière dont la simulation des trois étapes est effectuée avec ces modifications tout en tenant compte des modélisations géométriques détaillées de l'ADN est explicitée dans le chapitre 3 de ce manuscrit. Dans ce chapitre, la simulation est décrite comme étant une chaîne de calcul constituée de plusieurs modules devant se succéder pour réaliser une simulation complète. Cette architecture permet à la fois une grande flexibilité et la mise en oeuvre d'optimisations pour accélérer la simulation. Il s'agit par exemple de la possibilité d'exécuter la simulation sur plusieurs unités de calcul (parallélisme). Les hypothèses et paramètres associés à cette chaîne de calcul ainsi que les différents résultats obtenus ont été également introduits dans ce chapitre. Ajoutons que les sections efficaces des composants de l'ADN, récemment mesurées et analysées dans le cadre du projet européen BioQuaRT [BAEK et al., 2013, 2012; BUG, 2014] auquel l'IRSN a participé, ont été incluses dans Geant4.LDRI.01 avec un nouveau système de gestion des matériaux. Ces sections efficaces, introduites sous forme de tables de données, sont destinées à être rendues publiques au sein du code Geant4. Il en est de même pour le système de gestion des matériaux qui sera proposé à la collaboration Geant4. Ce nouveau système a permis la définition d'un nouveau matériau composite représentatif de l'ADN pouvant être utilisé lors du transport des particules dans les modélisations

détaillées de l'ADN issues de DnaFabric. L'ensemble de la simulation ainsi développée a été testé en reproduisant l'irradiation d'un noyau de cellule fibroblaste par des protons de différentes énergies. Les résultats ainsi obtenus concernent différents types de dommages à l'ADN et nous nous sommes particulièrement intéressés aux cassures simple et double brin dans ce manuscrit. Ainsi, les résultats obtenus sont présentés et discutés dans le chapitre 4.1 et des comparaisons sont effectuées avec les résultats expérimentaux disponibles [BELLI et al., 2001, 2000; CAMPA et al., 2005; FRANKENBERG et al., 1999] ainsi qu'avec les résultats issus de simulations similaires [FRIEDLAND et al., 2003; NIKJOO et al., 2001]. Les comparaisons effectuées avec les résultats expérimentaux ont permis de fixer certains des paramètres associés aux hypothèses incluses dans notre simulation. Finalement, les résultats générés par notre chaîne de calcul sont en bon accord avec les résultats expérimentaux et ceux obtenus avec d'autres codes de simulation. L'introduction du matériau composite représentatif de l'ADN lors du transport des particules (étape physique) occasionne une augmentation significative du nombre de dommages à l'ADN simulés (SSB et DSB). Toutefois, l'ampleur de cette influence varie en fonction du critère utilisé pour déterminer l'apparition des SSB_{direct} qui sont les dommages à l'ADN générés durant l'étape physique. De manière plus générale, le chapitre 4.1 montre que l'usage de différents critères influe significativement sur la quantité de dommages à l'ADN simulée et que ce paramètre doit être choisi avec soin. Un critère de sélection permettant l'obtention de résultats proches des mesures expérimentales est proposé par défaut dans notre simulation. Il s'agit d'un seuil en énergie déposée dans le squelette d'un nucléotide : si ce seuil est supérieur à 17,5 eV alors un SSB_{direct} est enregistré comme suggéré par des travaux expérimentaux [LOBACHEVSKY et MARTIN, 1996; PRISE et al., 2000]. D'autres hypothèses incluses dans notre simulation ont été testées : le fait de limiter la durée de l'étape chimique à 2,5 ns et de découper la chromatine en voxels. La limitation de la durée de l'étape chimique entraîne une sous-estimation de 30% du nombre de DSB simulé par rapport au nombre obtenu lorsqu'elle se déroule sans interruption jusqu'à 1 ps. Dans notre travail, cette diminution a été choisie pour représenter l'effet dit de "scavenging" des radicaux OH[•]. Ce effet "scavenging" consiste à prendre en compte le fait que, dans un noyau cellulaire, les radicaux OH[•] sont absorbés par différents éléments non représentés dans notre simulation (protéines par exemple). Le découpage de la chromatine en voxels n'a pas d'effet significatif sur les résultats, comme démontré dans le chapitre 4.1, mais permet de réduire très significativement la mémoire requise et le temps de calcul (1 semaine avec le découpage en voxels). Finalement, l'ensemble des hypothèses caractérisant notre simulation forme un tout cohérent qui permet de reproduire les résultats expérimentaux actuellement disponibles dans la littérature concernant les DSB générés suite à l'irradiation de cellules fibroblaste par des protons mono-énergétiques. Notons que les résultats du chapitre 4.1 concernent les protons et, dans une moindre mesure, les ions alpha. Une vérification des résultats produits par la simulation avec des ions alpha en tant que particules incidentes serait nécessaire mais les nombreux développements requis pour construire l'outil de simulation n'ont pas permis de dégager suffisamment de temps pour cela lors de ce travail de thèse, néanmoins, cette étude sera effectuée par la suite.

Les différentes méthodes mises en oeuvre par la chaîne de calcul développée dans ce travail ouvrent de nombreuses perspectives à différents niveaux. Tout d'abord, la création de DnaFabric résulte d'un mélange de technologies, concepts et représentations propres à l'informatique graphique avec ceux couramment utilisés en physique. La simulation permettant de créer une version "relaxée" du génome contenue dans un noyau en est un bon exemple. En effet, les règles définissant la simulation sont issues de la physique et

de la chimie tandis que la façon dont elles sont appliquées aux objets géométriques (ADN) est inspirée des méthodes utilisées en informatique graphique. Concrètement, au cours du processus de relaxation, les domaines qui constituent la chromatine sont déplacés lors de chaque "pas" effectué par la simulation. La direction et la longueur de chaque déplacement sont calculés à partir de principes physiques et chimiques implémentés dans la simulation, ces principes constituant les "règles" de la simulation. Ensuite, une fois ces calculs effectués, le domaine concerné est sélectionné et déplacé à l'aide de méthodes prédéfinies s'appuyant sur l'organisation des objets au sein de la modélisation géométrique de la chromatine. Cette organisation est spécifiquement conçue pour que les objets qui la constituent soient aisément accessibles et manipulables de manière à pouvoir les afficher rapidement sur l'écran. Au final, l'organisation des objets constituant la chromatine est inspirée de l'informatique graphique car créée pour l'affichage en 3D de l'ADN mais elle facilite également le déroulement de la simulation de relaxation en raison de la flexibilité qu'elle procure. Remarquons d'un point de vue plus général que le fait d'afficher des objets en 3D est un calcul en soit et, ainsi, il n'est pas suprenant que ce qui facilite l'affichage facilite également l'exécution d'autres types de calculs. A ce propos, il est intéressant de remarquer que c'est cette organisation des objets constituant la chromatine qui rend également possible la gestion de plusieurs millions de voxels lors du remplissage. Ajoutons que cette architecture permet aussi une mise à jour quasi-instantanée de la position du domaine sur l'écran rendant ainsi possible le suivi de la simulation en temps réel. L'utilisation de concepts issus de différents domaines au sein d'un même outil informatique pourrait être approfondie pour affiner la modélisation géométrique de l'ADN en intégrant des paramètres biologiques, cela pourrait permettre la prise en compte de différents types de chromatine par exemple (hétérochromatine et euchromatine). L'amélioration des capacités de visualisation en temps réel renforcerait la capacité de vérification rapide des nouvelles modélisations géométriques ainsi créées. Enfin, la prise en compte d'actions correctives de la part de l'utilisateur rendrait possible l'ajustement de certains éléments a posteriori, évitant ainsi la réalisation d'une longue et nouvelle simulation de relaxation de la chromatine par exemple. Cette logique transversale est également applicable à propos de la chaîne de calcul et de la simulation des étapes physique, physico-chimique et chimique. En effet, ces différentes étapes correspondent à des disciplines distinctes caractérisées par des échelles temporelles différentes. L'organisation de la simulation en "chaîne" constitue une manière d'interconnecter ces différentes échelles en effectuant les calculs physiques, qui concernent les processus les plus rapides, dans un premier temps puis les calculs chimiques dans un second. Cependant, la réalisation durant l'étape chimique de calculs impliquant des échelles temporelles courtes et proches de celles de l'étape physique apporterait un moyen d'affiner la simulation. Cela concernerait, notamment, la prise en compte de débits de dose car, dans certaines situations, l'irradiation peut être longue et entraîner un mélange des étapes physique et chimique dans le temps. Cette situation apparaît, notamment, dans le cas des irradiations effectuées avec des photons. Une autre application associée à la combinaison de différentes échelles temporelles serait de permettre la modélisation durant l'étape chimique de l'impact d'une réaction OH^{\bullet} -désoxyribose sur la structure de l'ADN à l'aide de la dynamique moléculaire [ABOLFATH et al., 2013]. La modélisation ainsi effectuée déterminerait alors la probabilité d'apparition d'une cassure de la chaîne ADN et remplacerait la probabilité fixe de 42% qui est utilisée dans notre travail. Une telle modélisation allongerait drastiquement la durée des simulations mais un usage "raisonné" de ces calculs peut être imaginé : une base de données serait peu à peu construite avec les résultats des calculs de dynamique moléculaire effectués dans le but de les réutiliser directement par la suite. Finale-

ment, les combinaisons d'échelles temporelles mentionnées précédemment reviennent à affiner la granularité temporelle et spatiale de la simulation en considérant de manière plus détaillée des processus tels que l'apparition des cassures à l'ADN. A l'opposé de cette démarche, l'usage d'une granularité plus grossière offrirait également des avantages significatifs tels qu'une plus grande rapidité de calcul ou consommation de mémoire réduite. Par exemple, l'utilisation lors de l'étape chimique d'une représentation basée sur des concentrations d'espèces chimiques [GILLESPIE, 1976] au lieu de molécules individuelles réduirait la mémoire requise par la simulation. Toutefois, cela impliquerait une perte de précision "spatiale" car les réactions entre espèces chimiques, notamment avec l'ADN, ne pourraient plus être "positionnées" dans l'espace. Ce type de représentations pourrait permettre d'étendre la simulation pour, par exemple, prendre en compte d'éléments autres que l'ADN au sein du noyau cellulaire (paroi cellulaire, mitochondrie...) ou encore des tissus cellulaires. A ce propos, des travaux [KLANN et al., 2012; WYLIE et al., 2006] montrent qu'il est possible de créer des simulations hybrides capables de combiner de manière harmonieuse différentes représentations (molécules individuelles et concentrations) et donc différents niveaux de granularité de l'espace. De telles simulations cumulent les avantages des niveaux de granularité considérés, soit généralement une grande précision pour une granularité fine et une grande rapidité pour une granularité grossière.

Les premiers résultats obtenus avec notre chaîne de calcul concernent essentiellement le noyau de fibroblaste humain irradié par des protons. Des résultats similaires mais impliquant différents noyaux cellulaires et particules primaires pourraient être générés de manière à améliorer l'équilibre de la simulation et finaliser l'ajustement de ces paramètres. Notons que l'utilisation de photons en tant que particules primaires constitue un cas particulier du point de vue de la simulation car un très grand nombre de photons doivent être simulés et leurs effets combinés pour reproduire les conditions réelles d'une irradiation et d'un débit de dose. La réalisation d'une telle simulation avec notre chaîne de calcul est possible mais nécessite toutefois d'effectuer quelques modifications qui seront faites par la suite. Il est également envisagé d'étendre la chaîne de calcul par l'ajout de nouveaux modules afin de prendre en compte des processus succédant à l'apparition des dommages à l'ADN, tels que certains processus de réparation de l'ADN par exemple. Notons que de tels ajouts peuvent être effectués dans des cadres autres que l'approche mécanistique choisie pour ce travail. En effet, des modèles paramétriques pourraient être avantageusement mis en place, la rapidité de calcul associée à ces modèles leur conférant un intérêt certain. Dans l'ensemble, l'extension de la simulation est particulièrement importante car elle permet de s'approcher des effets biologiques détectés dans le cadre du projet ROSIRIS notamment. A cet effet, une prochaine thèse est justement prévue à l'IRSN pour étendre la simulation construite dans ce travail et réaliser ce rapprochement.

Références

- ABOLFATH, R. M., D. J. CARLSON, Z. J. CHEN et R. NATH. 2013, «A molecular dynamics simulation of DNA damage induction by ionizing radiation», *Physics in medicine and biology*, vol. 58, n° 20, p. 7143. [171](#)
- BAEK, W. Y., A. ARNDT, M. U. BUG, H. RABUS et M. WANG. 2013, «Total electron-scattering cross sections of pyrimidine», *Physical Review A*, vol. 88, n° 3. [169](#)
- BAEK, W. Y., M. BUG, H. RABUS, E. GARGIONI et B. GROSSWENDT. 2012, «Differential elas-

- tic and total electron scattering cross sections of tetrahydrofuran», *Physical Review A*, vol. 86, n° 3. 169
- BELLI, M., R. CHERUBINI, M. DALLA VECCHIA, V. DINI, G. ESPOSITO, G. MOSCHINI, O. SAPORA, C. SIGNORETTI, G. SIMONE, E. SORRENTINO et M. TABOCCHINI. 2001, «DNA Fragmentation in mammalian cells exposed to various light ions», *Advances in Space Research*, vol. 27, n° 2, p. 393–399. 170
- BELLI, M., R. CHERUBINI, M. DALLA VECCHIA, V. DINI, G. MOSCHINI, C. SIGNORETTI, G. SIMONE, M. A. TABOCCHINI et P. TIVERON. 2000, «DNA DSB induction and rejoining in V79 cells irradiated with light ions: a constant field gel electrophoresis study», *International Journal of Radiation Biology*, vol. 76, n° 8, p. 1095–1104. 170
- BUG, M. U. 2014, *Nanodosimetric particle track simulations in water and DNA media*, thèse de doctorat, University of Wollongong, School of Physics. 169
- CAMPA, A., F. BALLARINI, M. BELLI, R. CHERUBINI, V. DINI, G. ESPOSITO, W. FRIEDLAND, S. GERARDI, S. MOLINELLI, A. OTTOLENGHI, H. PARETZKE, G. SIMONE et M. A. TABOCCHINI. 2005, «DNA DSB induced in human cells by charged particles and gamma rays: Experimental results and theoretical approaches», *International Journal of Radiation Biology*, vol. 81, n° 11, p. 841–854. 170
- FRANKENBERG, D., H. J. BREDE, U. J. SCHREWE, C. STEINMETZ, M. FRANKENBERG-SCHWAGER, G. KASTEN et E. PRALLE. 1999, «Induction of DNA Double-Strand Breaks by 1h and 4he Ions in Primary Human Skin Fibroblasts in the LET Range of 8 to 124 keV/um», *Radiation Research*, vol. 151, p. 540–549. 170
- FRIEDLAND, W., P. JACOB, P. BERNHARDT, H. G. PARETZKE et M. DINGFELDER. 2003, «Simulation of DNA damage after proton irradiation», *Radiation Research*, vol. 159, p. 401–410. 170
- GILLESPIE, D. T. 1976, «A general method for numerically simulating the stochastic time evolution of coupled chemical reactions», *Journal of computational physics*, vol. 22, n° 4, p. 403–434. 172
- KLANN, M., A. GANGULY et H. KOEPL. 2012, «Hybrid spatial Gillespie and particle tracking simulation», *Bioinformatics*, vol. 28, n° 18, p. i549–i555. 172
- LOBACHEVSKY, P. N. et R. F. MARTIN. 1996, «DNA Strand Breakage by ¹²⁵I-Decay in a Synthetic Oligodeoxynucleotide: Quantitative analysis of fragment distribution», *Acta Oncologica*, vol. 35, n° 7, p. 809–815. 170
- MEYLAN, S., U. VIMONT, S. INCERTI, I. CLAIRAND et C. VILLAGRASA. 2016, «Geant4-DNA simulations using complex DNA geometries generated by the DnaFabric tool», *Computer Physics Communications*, vol. 204, p. 159–169. 169
- NIKJOO, H., P. O'NEILL, W. E. WILSON et D. T. GOODHEAD. 2001, «Computational approach for determining the spectrum of DNA damage induced by ionizing radiation», . 170
- PRISE, K. M., M. FOLKARD, B. D. MICHAEL, B. VOJNOVIC, B. BROCKLEHURST, A. HOPKIRK et I. H. MUNRO. 2000, «Critical energies for SSB and DSB induction in plasmid DNA by

low-energy photons : action spectra for strand-break induction in plasmid DNA irradiated in vacuum», *International journal of radiation biology*, vol. 76, n° 7, p. 881–890. [170](#)

WYLIE, D. C., Y. HORI, A. R. DINNER et A. K. CHAKRABORTY. 2006, «A Hybrid Deterministic Stochastic Algorithm for Modeling Cell Signaling Dynamics in Spatially Inhomogeneous Environments and under the Influence of External Fields», *The Journal of Physical Chemistry B*, vol. 110, n° 25, p. 12 749–12 765. [172](#)

Annexe A

Sections efficaces de l'ADN pour les électrons

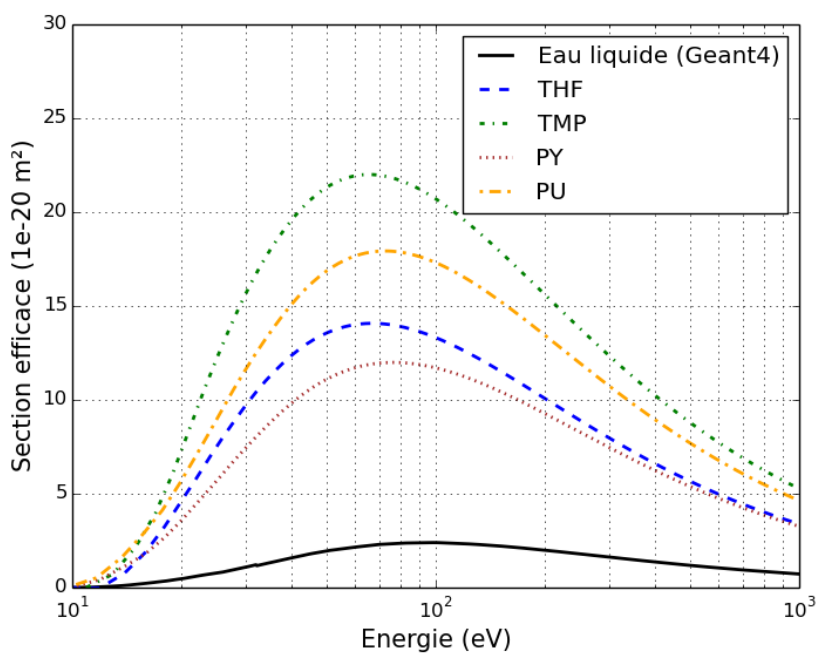


FIGURE A.1 – Sections efficaces totales d'ionisation pour les électrons et les matériaux précurseurs THF, TMP, PY et PU. Les sections efficaces totales d'ionisation de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.

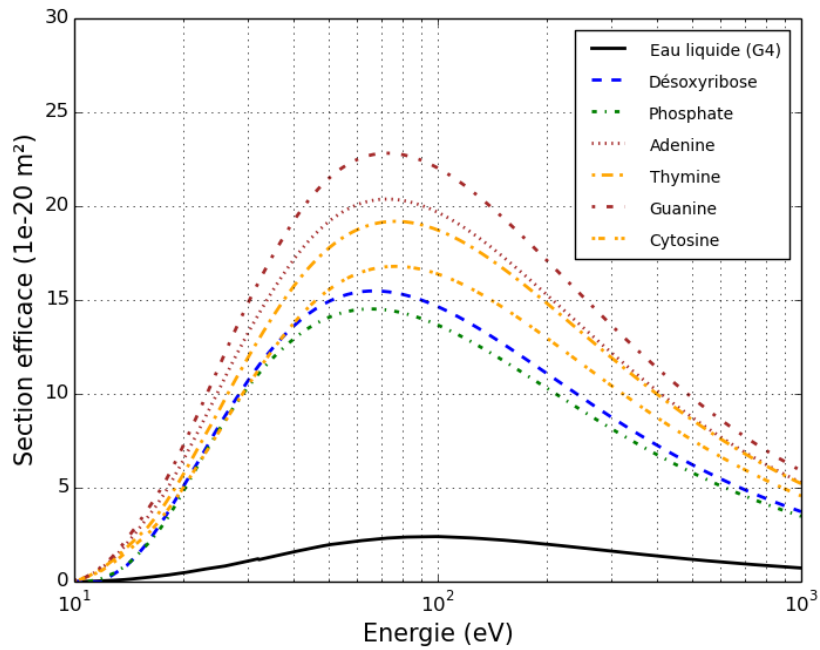


FIGURE A.2 – Sections efficaces totales d'ionisation pour les matériaux ADN : désoxyribose, phosphate, adénine, thymine, guanine et cytosine. Les sections efficaces totales d'ionisation de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.

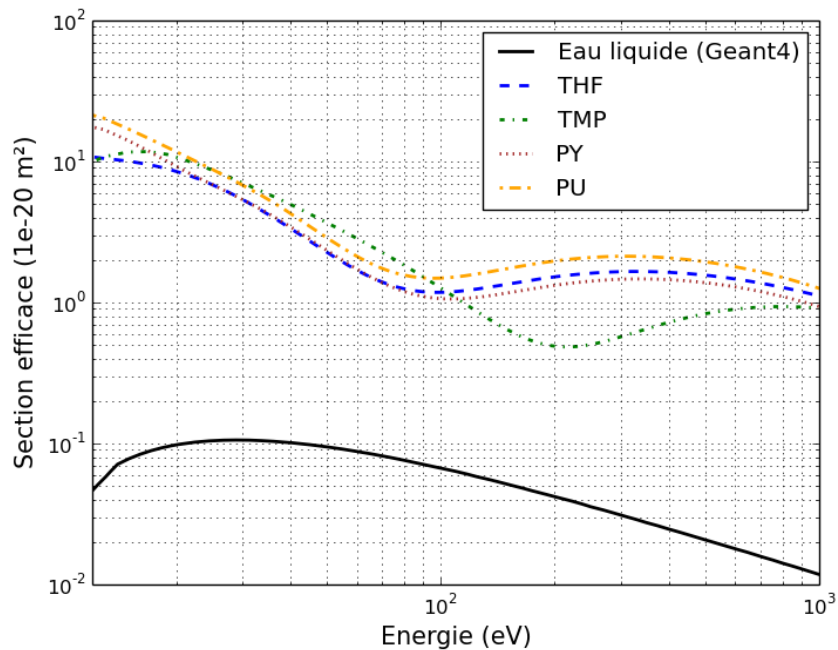


FIGURE A.3 – Sections efficaces totales d'excitation pour les électrons et les matériaux précurseurs THF, TMP, PY et PU. Les sections efficaces totales d'excitation de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.

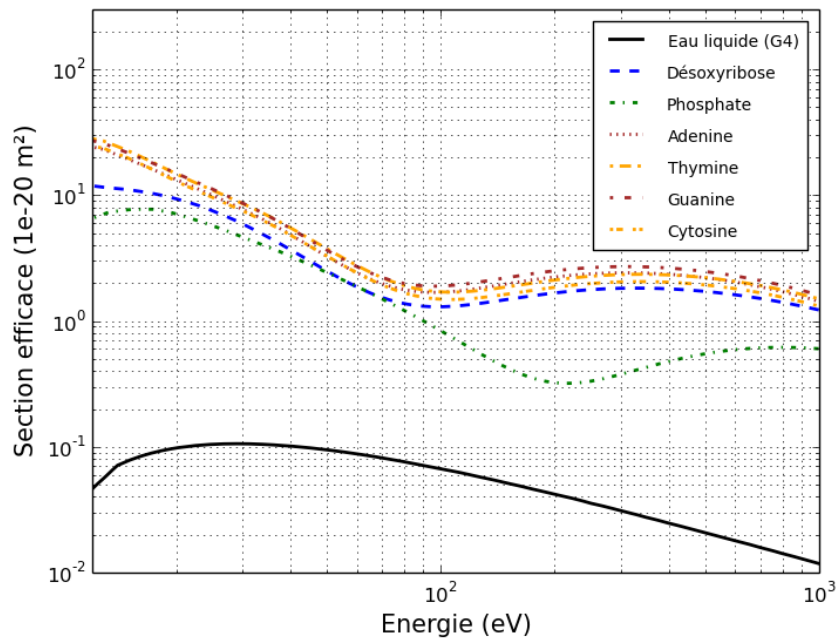


FIGURE A.4 – Sections efficaces totales d'excitation pour les électrons et les matériaux ADN : désoxyribose, phosphate, adénine, thymine, guanine et cytosine. Les sections efficaces totales d'excitation de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.

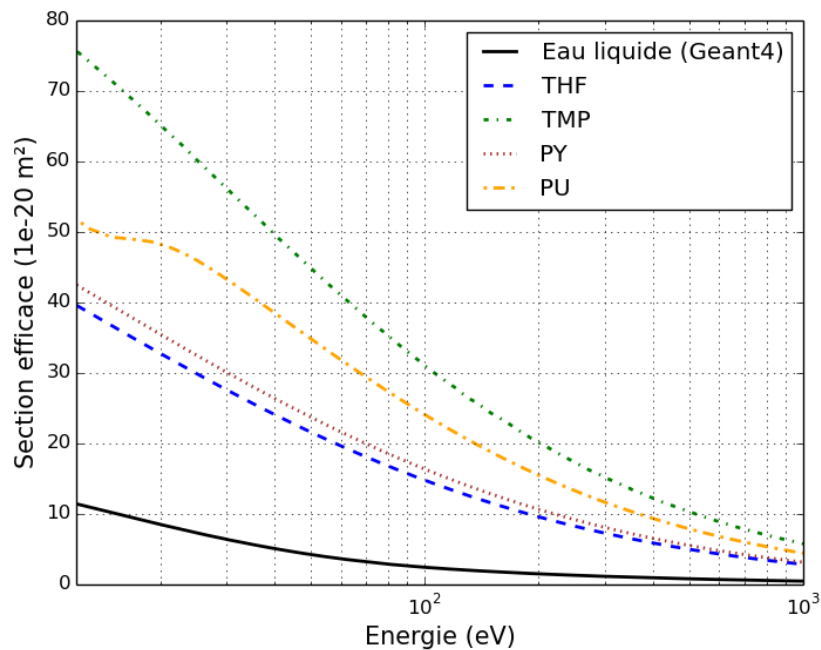


FIGURE A.5 – Sections efficaces totales de diffusion élastique pour les électrons et les matériaux précurseurs THF, TMP, PY et PU. Les sections efficaces totales de diffusion élastique de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.

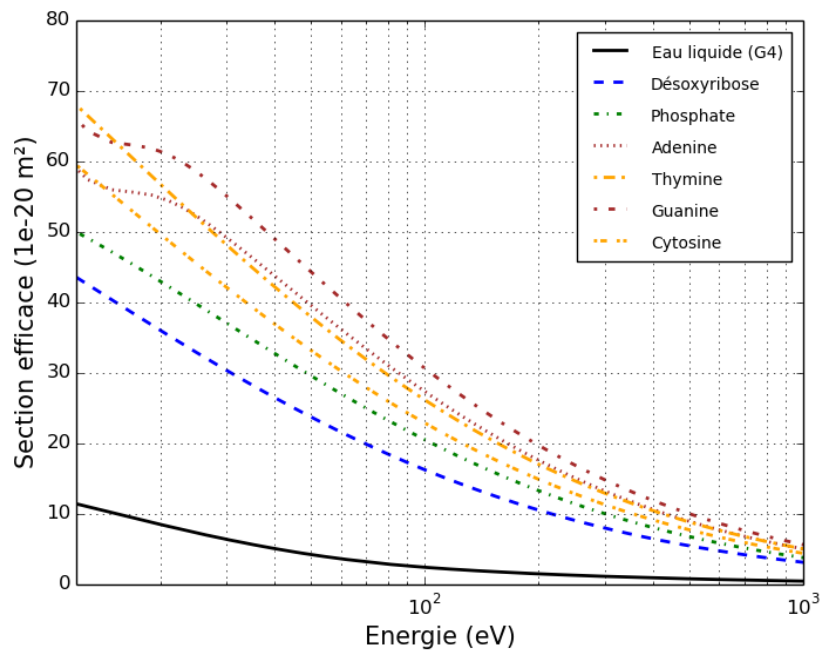
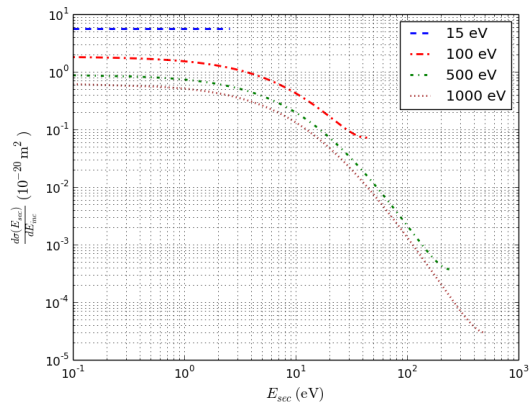
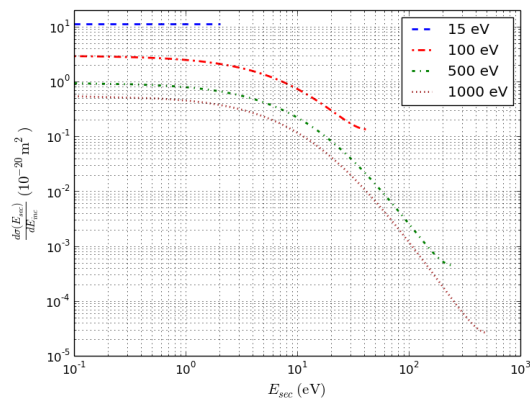


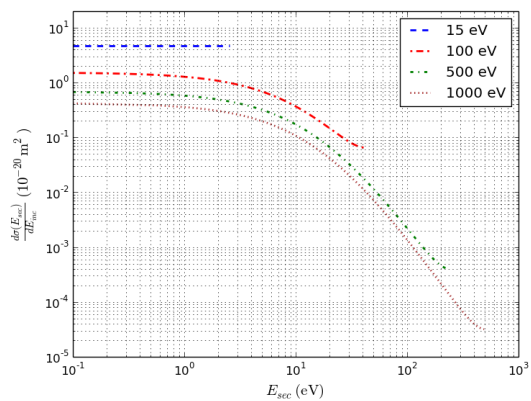
FIGURE A.6 – Sections efficaces totales de diffusion élastique les électrons et les matériaux ADN : désoxyribose, phosphate, adénine, thymine, guanine et cytosine. Les sections efficaces totales de diffusion élastique de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.



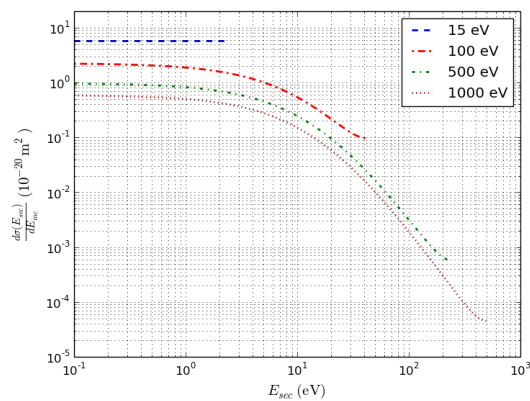
(a) Tetrahydrofurane



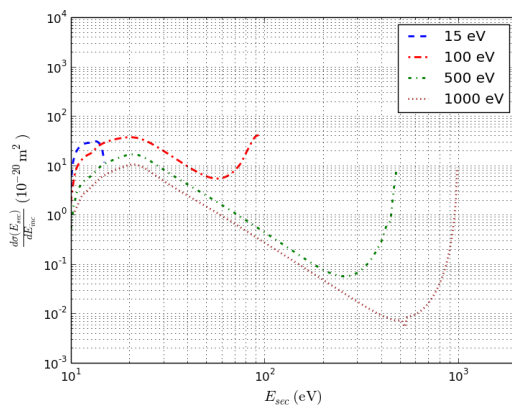
(b) Timethylphosphate



(c) Pyrimidine

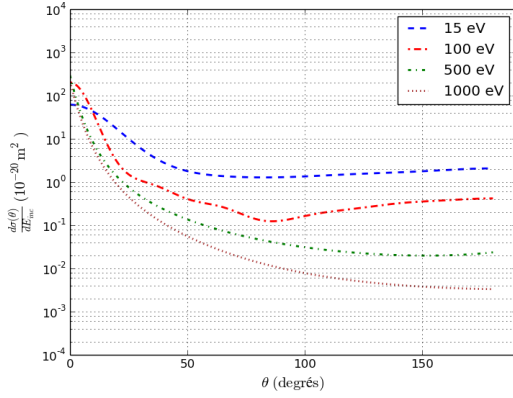


(d) Purine

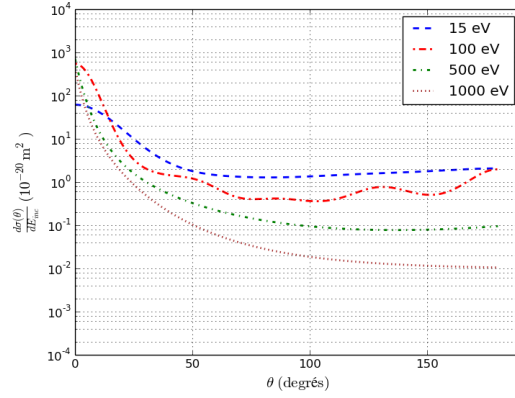


(e) Eau liquide (Geant4-DNA - Geant4.10.1)

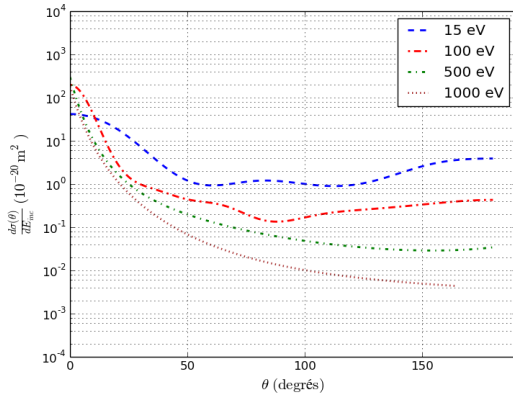
FIGURE A.7 – Sections efficaces différentielles d’ionisation pour les électrons et les matériaux pré-curseurs THE, TMP, PY et PU. Les sections efficaces différentielles d’ionisation concernant l’eau liquide (Geant4.10.01) sont également présentées. Pour chaque matériau, quatre énergies (E_{inc}) de particule incidente sont considérées : 15, 100, 500 et 1000 eV. La variation des sections efficaces différentielles est illustrée en fonction de l’énergie (E_{sec}) de la particule secondaire émise.



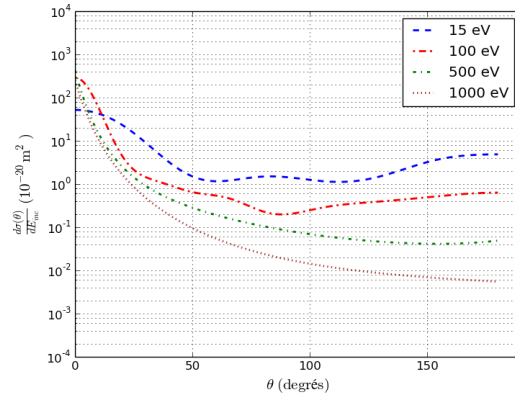
(a) Tetrahydrofurane



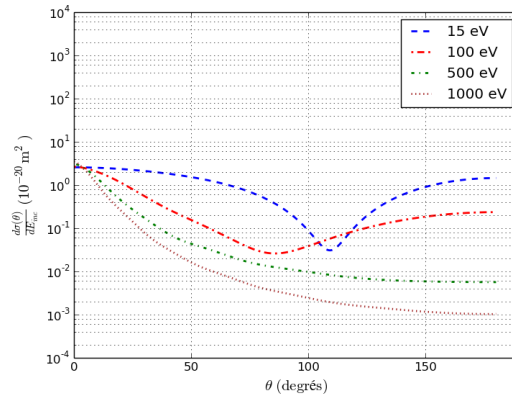
(b) Timethylphosphate



(c) Pyrimidine



(d) Purine



(e) Eau liquide (Geant4-DNA - Geant4.10.1)

FIGURE A.8 – Sections efficaces différentielles de diffusion élastique pour les électrons et les matériaux précurseurs THE, TMP, PY et PU. Les sections efficaces différentielles de diffusion élastique concernant l'eau liquide (Geant4.10.01) sont également présentées. Pour chaque matériau, quatre énergies (E_{inc}) de particule incidente sont considérées : 15, 100, 500 et 1000 eV. La variation des sections efficaces différentielles est illustrée en fonction de l'angle θ de déviation de la particule incidente (vis à vis de sa trajectoire initiale).

Annexe B

Sections efficaces de l'ADN pour les protons

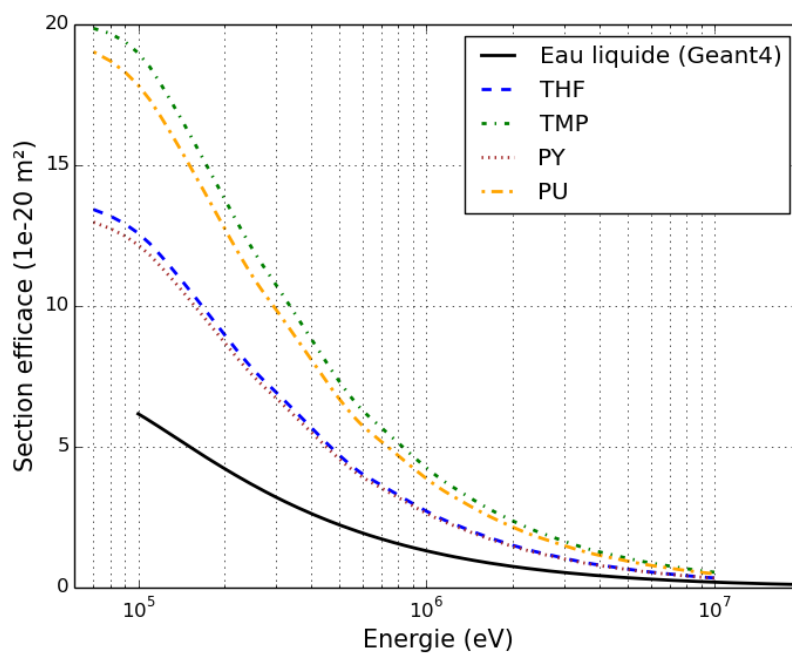


FIGURE B.1 – Sections efficaces totales d'ionisation pour les protons et les matériaux précurseurs THF, TMP, PY et PU. Les sections efficaces totales d'ionisation de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.

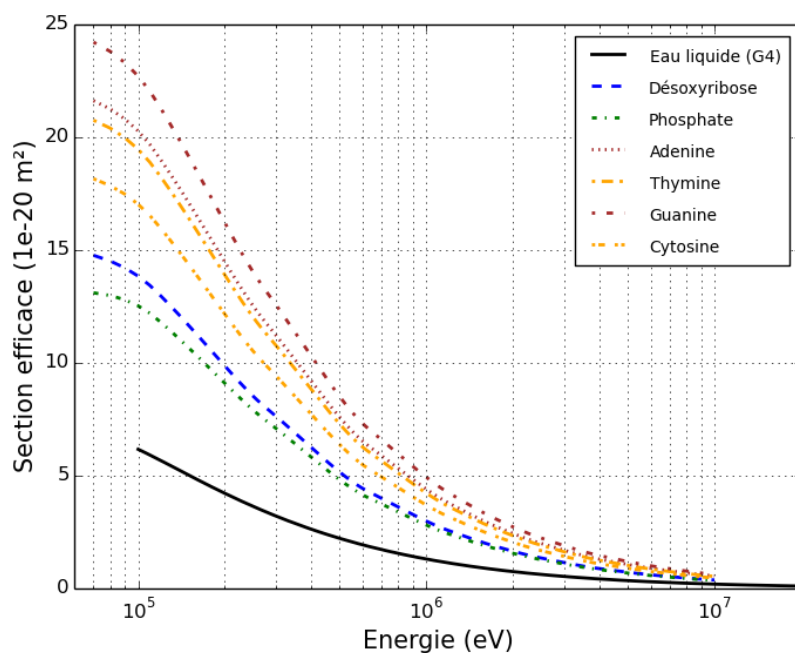
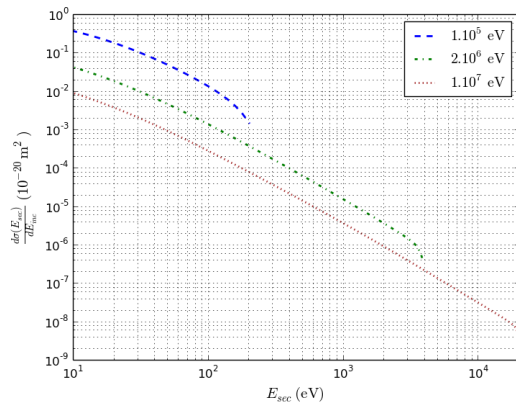
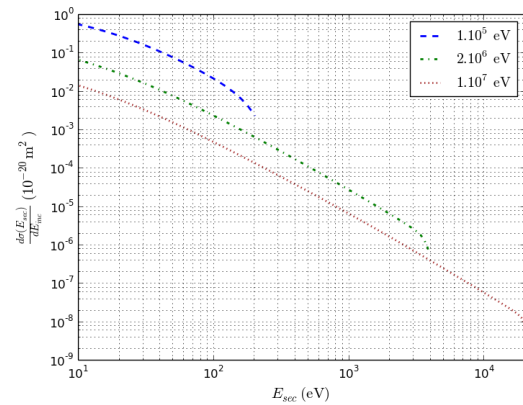


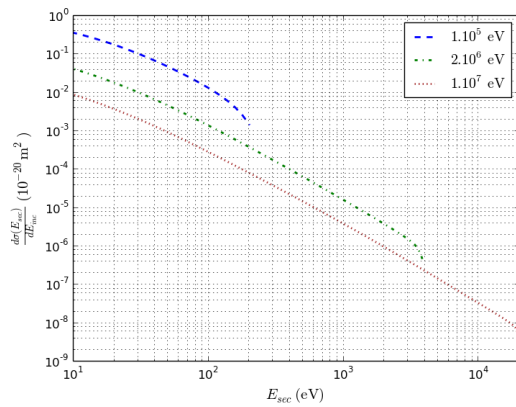
FIGURE B.2 – Sections efficaces totales d'ionisation pour les protons et les matériaux ADN : désoxyribose, phosphate, adénine, thymine, guanine et cytosine. Les sections efficaces totales d'ionisation de l'eau liquide utilisées dans Geant4.10.01 sont également présentées.



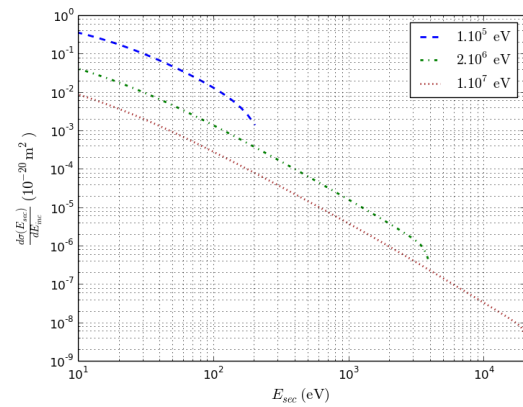
(a) Tetrahydrofurane



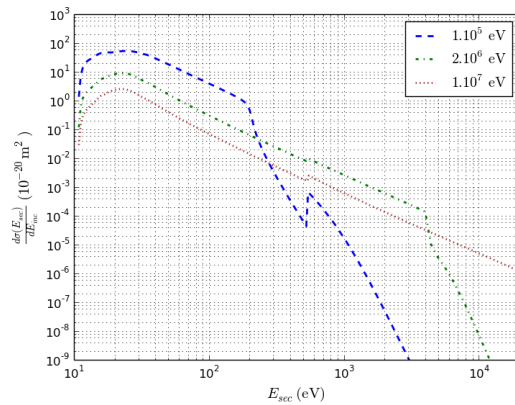
(b) Timethylphosphate



(c) Pyrimidine



(d) Purine

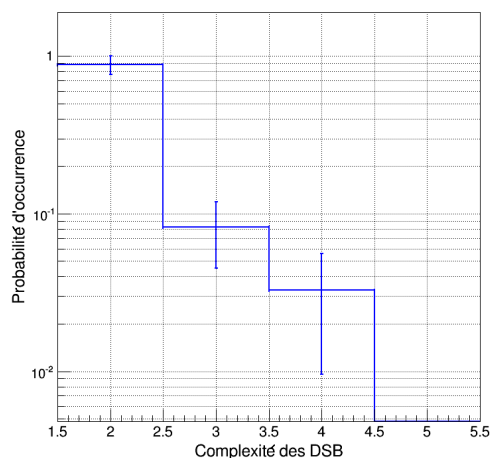


(e) Eau liquide (Geant4-DNA - Geant4.10.1)

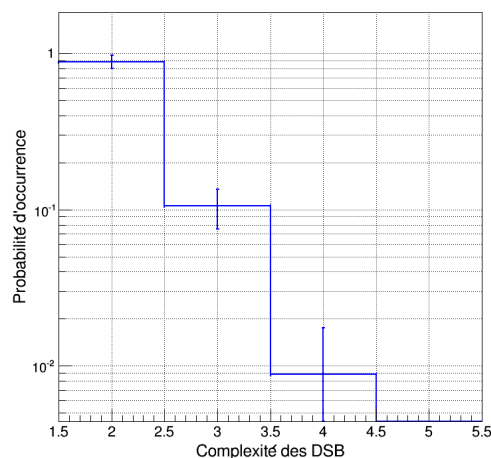
FIGURE B.3 – Sections efficaces différentielles d’ionisation pour les protons et les matériaux pré-curseurs THE, TMP, PY et PU. Les sections efficaces différentielles d’ionisation concernant l’eau liquide (Geant4.10.01) sont également présentées. Pour chaque matériau, quatre énergies (E_{inc}) de particule incidente sont considérées : $1 \cdot 10^5$, $2 \cdot 10^6$, $1 \cdot 10^7$ eV. La variation des sections efficaces différentielles est illustrée en fonction de l’énergie E_{sec} de la particule incidente.

Annexe C

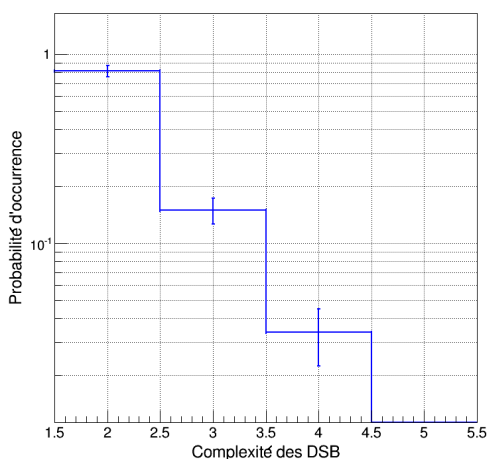
Distributions de la complexité des DSB simulés avec des protons primaires



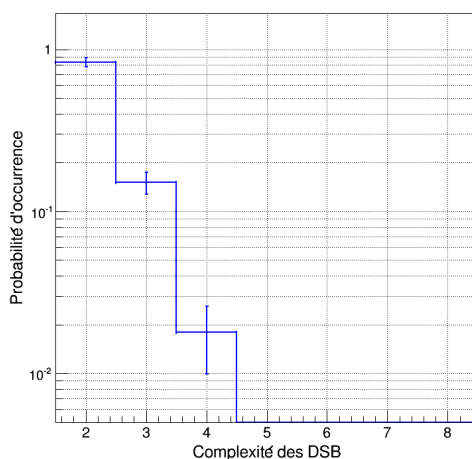
(a) 20 MeV (TEL 2,8 keV/ μm).



(b) 10 MeV (TEL 5 keV/ μm).

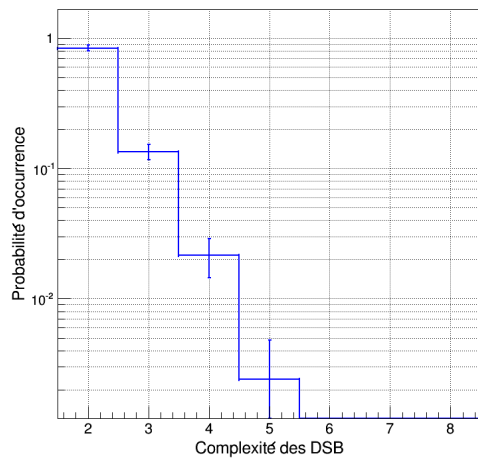


(c) 5 MeV (TEL 8,7 keV/ μm).

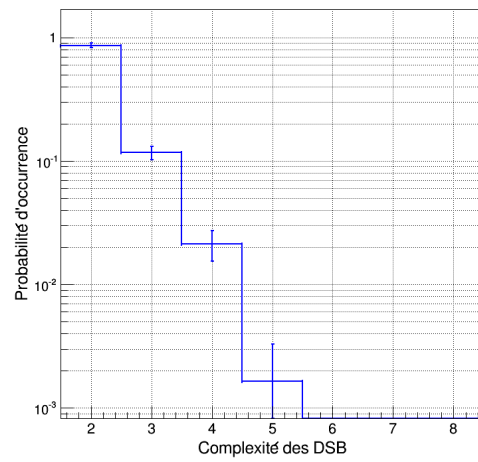


(d) 4 MeV (TEL 10,3 keV/ μm).

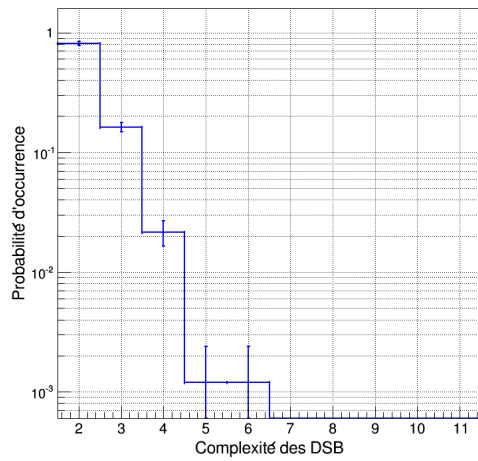
FIGURE C.1 – Distribution de la complexité des DSB simulés avec FullSim dans un noyau de cellule fibroblaste pour différentes énergies de protons primaires : 20, 10, 5 et 4 MeV. La simulation est dans sa configuration par défaut.



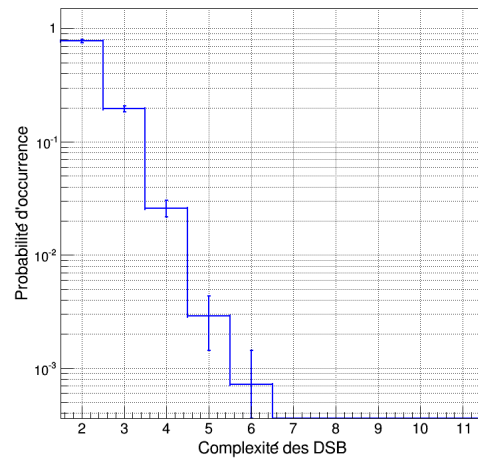
(a) 3 MeV (TEL 12,8 keV/μm).



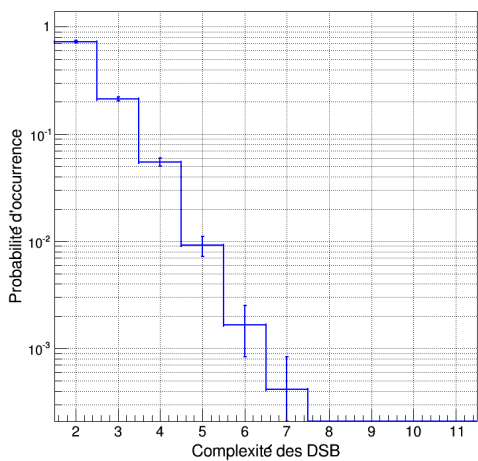
(b) 2 MeV (TEL 17,4 keV/μm).



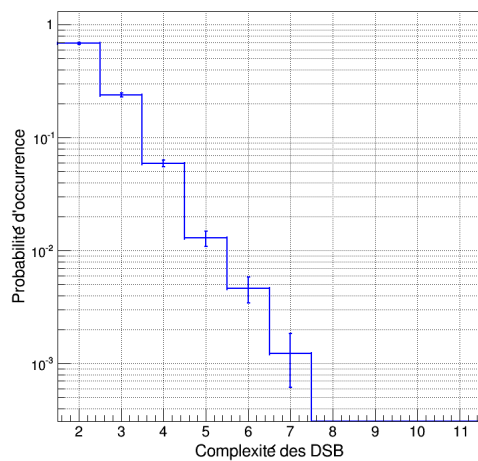
(c) 1,5 MeV (TEL 21,3 keV/μm).



(d) 1 MeV (TEL 28,9 keV/μm).



(e) 0,7 MeV (TEL 37,6 keV/μm).



(f) 0,5 MeV (TEL 42,6 keV/μm).

FIGURE C.2 – Distribution de la complexité des DSB simulés avec FullSim dans un noyau de cellule fibroblaste pour différentes énergies de protons primaires : 3 ; 2 ; 1,5 ; 1 ; 0,7 et 0,5 MeV. La simulation est dans sa configuration par défaut.

