



Université d'Evry Val d'Essonne
Filière : L3ASR

<p>Auto Assemblage d'ADN : aTAM VS Grammaires de tableaux</p>

Rapport de stage effectué au laboratoire ibisc
présenté par :
SI KADDOUR Mustapha
durée : 2 mois

Encadré par :
prof. Sergui IVANOV

Année universitaire : 2021-2022

Remerciements

Tout d'abord, je remercie très vivement M. Sergui IVANOV pour m'avoir encadré et conseillé au cours de mon stage et pour m'avoir consacré de son temps.

Je remercie ensuite tous les membres de l'équipe pour leur accueil très chaleureux et les autres stagiaires pour tous les bons moments passés ensemble.

Enfin, je remercie grandement tout le personnel du laboratoire ibisc pour leur engagement et leur services afin qu'on puisse travailler dans les meilleures conditions possible.

Table des matières

1	Introduction	4
2	Annonce de plan	5
3	Présentation du IBISC	5
4	L'objet de ce stage et les différentes missions effectuées	6
5	Documents traités	6
6	Résumé du travail effectué	6
6.1	Modèle abstraits d'auto assemblage	6
6.2	Rappel : Grammaire de chaînes de caractères	7
6.3	Grammaire de tableaux	8
6.3.1	Derivation modes	8
6.3.2	Exemple	9
6.4	Conversion aTAM \rightarrow grammaire de tableaux	10
6.4.1	Preuve	10
7	Conclusion	11
8	Bilan de stage	12
8.1	Compétences acquises	12
8.2	Difficultés rencontrées et leur résolution	12

1 Introduction

L'ADN se présente sous la forme d'un assemblage en double hélice composée de deux chaînes nucléiques complémentaires appelées brins. Les sous-unités qui forment ces chaînes sont des nucléotides dont la composition est un groupement phosphate lié à un sucre (désoxyribose), lui-même lié à une base azotée qui peut être de quatre types : Adénine (A), Guanine (G), Cytosine (C), et Thymine (T). Il est le support de l'hérédité ou bien l'information génétique, c'est à dire qu'il porte des milliers de gènes sous la forme de succession de nucléotides.

L'assemblage des atomes ou des molécules individuelles détermine les propriétés physiques et chimiques d'un matériau. Si ces propriétés peuvent être manipulées avec précision, une variété de matériaux précieux avec de nouvelles caractéristiques peuvent être synthétisées et adaptées spécifiquement à une application donnée. Cependant, jusqu'à présent, cela reste un défi majeur dans les domaine de la chimie et de la science des matériaux. La nanotechnologie de l'ADN offre la possibilité de personnaliser ces propriétés.

Dans les années 1980, le professeur Seeman a d'abord proposé d'utiliser des brins d'ADN pour s'auto-assembler de bas en haut afin de former des cristaux. Pour réaliser une synthèse et un contrôle raffinés plus complexes, plusieurs méthodes d'assemblage ont été proposées : en utilisant les tuiles d'ADN, les briques d'ADN et l'ADN origami.

Cette technologie vise à développer des composants et des systèmes à l'échelle nanométrique en grandes quantités et à un coût faible. Les méthodes traditionnelles reposent sur la commande externe de grandes machines pour extraire des substances en vrac afin de créer des motifs, ce sont des techniques puissantes mais elles sont entravées par leurs coûts cher et aussi par des limitations de taille.

Le modèle d'auto-assemblage le plus étudié est le modèle abstrait (aTAM), introduit par Winfree comme un modèle informatique d'auto-assemblage basé sur des tuiles d'ADN. Ce modèle est une version algorithmique des pavages de Wang et peut être considéré comme un automate cellulaire asynchrone.

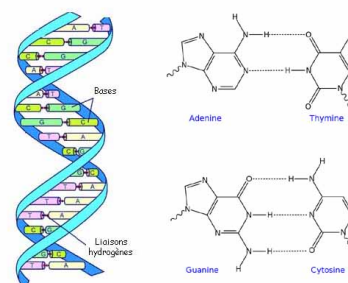


FIGURE 1 – La structure de la molécule d'ADN

Source: <https://www.police-scientifique.com/adn/structure-et-principe-de-base>

2 Annonce de plan

À travers ce rapport nous verrons dans un premier temps une description du laboratoire, puis l'objet de ce stage et les différentes missions effectuées, le travail réalisé et les résultats obtenus et par la fin un bilan général.

3 Présentation du IBISC

Le laboratoire IBISC (Informatique, BioInformatique, Systèmes Complexes EA 4526) est un laboratoire de l'Université d'Évry Paris-Saclay. Il est implanté sur deux sites de l'université : IBGBI et PELVOUX et rattaché à deux UFRs scientifiques : l'UFR Sciences Fondamentales et Appliquées (SFA) et l'UFR Sciences et Technologies (ST).

IBISC regroupe plusieurs enseignants-chercheurs (56 enseignants chercheurs permanents, 3 chercheurs associés, 4 BIATSS, 39 doctorants, 1 post-doc, on peut compter un effectif total de plus de 100 personnes en 2019.) de l'UEVE qui sont impliqués dans la gouvernance de l'université (7 Vice-Présidents, des membres élus au Conseil d'Administration, à la Commission de la Formation et de la Vie Universitaire, à la Commission de la Recherche, 1 directeur d'IUT, 1 directeur d'UFR, 8 directeurs de départements d'enseignement en 2017).

Ses activités de recherche s'articulent autour du traitement de la modélisation, la conception, la simulation et la validation des systèmes biologiques complexes, on peut trouver 4 principales thèmes :

1. Bioinformatique et systèmes biologiques.
2. Assistance aux personnes, signaux et images pour le biomédical.
3. Systèmes autonomes intelligents.
4. Systèmes ouverts et sûrs.

Ces thèmes sont gérés par 4 équipes de recherche : AROBAS, COSMO, IRA2 et SIAM.

L'IBISC développe des interactions avec d'autres laboratoires UEVE du secteur de la santé, affiliés à Génopole, ENSIEE et Télécom SudParis. Il développe aussi des liens avec ses UFRs de rattachement (SFA et ST) ce qui permet une gestion cohérente des ressources humaines et matérielles à travers des plateformes de recherche et d'enseignement (volière, robots, cube immersif 3D, etc.).

Outre les activités de santé, IBISC est également actif dans le domaine de la robotique aérienne et de la mobilité intelligente.

4 L'objet de ce stage et les différentes missions effectuées

L'**objectif** de ce stage est de comparer le modèle classique d'auto-assemblage de tuiles (Abstract Tile Assembly Model, aTAM) avec les grammaires de tableaux, qui sont un outil classique mais pas très utilisé des langages formels.

Différentes sortes d'activités m'ont été confiées :

1. Découvrir les motivations et les aspects pratique d'auto-assemblage d'ADN à travers la lecture des articles.
2. Découvrir et analyser le modèle aTAM à travers l'étude de l'article "The program-size complexity of self-assembled paths de Pierre-Étienne Meunier, Damien Regnault et Damien Woods".
3. Établir l'état de l'art des travaux pertinents autour des grammaires de tableaux.
4. Poser le théorème de la conversion aTAM \rightarrow Grammaires de tableaux, et le prouver.
5. Apprendre le langage de programmation Racket et Réaliser un simulateur des grammaires de tableaux.

5 Documents traités

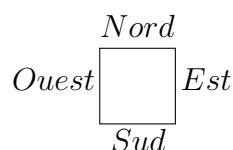
1. Algorithmic self assembly dna sierpinski triangles de Paul W. K. Rothemund, Nick Papadakis, Erik Winfree.
2. The program-size complexity of self-assembled paths de Pierre-Étienne Meunier, Damien Regnault et Damien Woods.
3. Array Insertion and Deletion P Systems de Henning Fernau, Rudolf Freund, Sergiu Ivanov, Markus L. Schmid, et K.G. Subramanian.

6 Résumé du travail effectué

6.1 Modèle abstraits d'auto assemblage

Le modèle abstrait d'assemblage de tuiles a été introduit par Winfree en 1998.

Definition 1. une tuile se représente sous forme d'un carré unitaire à quatre côtés, chacun composé d'un type de colle (liaisons d'hydrogène) qui peut être forte ou faible et d'une force entière non négative. Les côtés d'une tuile sont respectivement appelés nord, est, sud et ouest, comme illustré dans l'image suivante :



il existe deux types de ce modèle :

1. **Le modèle non coopératif** (de température 1) : Les tuiles peuvent s'attacher à l'assemblage si au moins le type de l'un de leurs côtés correspond à un type d'un côté de l'assemblage adjacent à la position où elles s'attachent (elles n'ont pas besoin d'attendre que d'autres liaisons apparaissent adjacentes à leur position d'attachement).
2. **Le modèle coopératif** (de temperature 2) : Aucune tuile ne peut être ajoutée à l'assemblage tant que les deux tuiles précédentes ne sont pas déjà présentes.
 - Il existe deux types de liens : forts et faibles. une tuile peut s'attacher à un assemblage par un côté si ce côté forme une liaison forte ("force 2") avec l'assemblage, ou bien si deux de ses côtés correspondent chacun avec une liaison faible ("force 1").
 - La fixation d'une tuile par deux colles faibles ne peut se produire qu'après la présence des deux tuiles voisines, donc le système sera obligé d'attendre plutôt que de continuer avec des informations fausses ou incomplètes.

Nous allons étudier la restriction de ce modèle donc **le modèle non coopératif**.

Definition 2. Un assemblage est une fonction $f : \mathbb{Z}^2 \rightarrow T$ où T est un ensemble de types de tuiles. On note \mathcal{A}^T l'ensemble de tous les assemblages sur l'ensemble de types de tuiles T . Deux types de tuiles dans un assemblage sont dits attachés de manière stable, si les types de colle sur leurs côtés attenants sont égaux et ont une force ≥ 1 .

Definition 3. Un système d'assemblage de tuiles est un triple $\mathcal{T} = (T, \sigma, 1)$ où T est un ensemble fini de types de tuiles, σ est un assemblage τ -stable appelé la graine et $\tau \in \mathbb{N}$ est la température. Tout au long de cet article, $\tau = 1$.

6.2 Rappel : Grammaire de chaînes de caractères

Definition 4. Une grammaire (séquentielle) G est un 5-uplet $(O, O_T, w, P, \Rightarrow_G)$ où O est un ensemble d'objets, $O_T \subseteq O$ est un ensemble d'objets terminaux, $w \in O$ est l'axiome (objet de départ), P est un ensemble fini de règles, et $\Rightarrow_G \subseteq O \times O$ est la relation de dérivation de G . Une règle $p \in P$ est dite applicable à un objet $x \in O$ si et seulement s'il existe au moins un objet $y \in O$ tel que $(x, y) \Rightarrow_p$, on écrit aussi $x \Rightarrow_p y$. La relation de dérivation \Rightarrow_G est l'union de tous les \Rightarrow_p , c'est-à-dire $\Rightarrow_G = \cup_{p \in P} \Rightarrow_p$. La clôture réflexive et transitive de \Rightarrow_G est notée \Rightarrow_G^* .

Definition 5. Une grammaire de chaînes de caractères GS est représentée par $((N \cup T)^*, T^*, w, P, \Rightarrow_p)$ où N est l'alphabet des symboles non terminaux, T est l'alphabet des symboles terminaux, $N \cap T = \emptyset$, $w \in (N \cup T)^+$ est l'axiome, P est un ensemble fini de règles de réécriture de chaînes, et la relation de dérivation \Rightarrow_{G_s} est la relation classique pour les grammaires de chaînes définies sur $V^* \times V^*$ avec $V := N \cup T$.

6.3 Grammaire de tableaux

Definition 6. Soit $d \in \mathbb{N}$, un tableau \mathcal{A} de dimension d sur un alphabet V est une fonction $\mathcal{A} : \mathbb{Z}^* \rightarrow V \cup \#$ où $shape(\mathcal{A}) = \{v \in \mathbb{Z} \mid \mathcal{A}(v) \neq \#\}$ est fini et $\# \notin V$ est appelé le fond ou le symbole vide. On écrit généralement $\mathcal{A} = \{(v, \mathcal{A}(v)) \mid v \in shape(\mathcal{A})\}$.

L'ensemble de tous les tableaux d -dimensionnels sur V est noté V^{*d} . Le tableau vide dans V^{*d} est noté Λ_d . De plus, on définit $V^{+d} = V^{*d} \setminus \{\Lambda_d\}$.

La translation $\mathcal{T}_v : \mathbb{Z}^d \rightarrow \mathbb{Z}^d$ est défini par $\mathcal{T}_v(w) = w + v$ pour tout $w \in \mathbb{Z}^d$.

Pour tout tableau $\mathcal{A} \in V^{*d}$ on définit $\mathcal{T}_v(\mathcal{A})$, le tableau d -dimensionnel translaté par v , par $(\mathcal{T}_v(\mathcal{A}))(w) = \mathcal{A}(w - v)$ pour tout $w \in \mathbb{Z}^d$.

La classe d'équivalence $[\mathcal{A}]$ d'un tableau $\mathcal{A} \in V^{*d}$ est définie par :

$$[\mathcal{A}] = \{\mathcal{B} \in V^{*d} \mid \mathcal{B} = \mathcal{T}_v(\mathcal{A}) \text{ pour } v \in \mathbb{Z}^d\}$$

L'ensemble de toutes les classes d'équivalence des tableaux d -dimensionnels sur V par rapport aux translations linéaires est noté $[V^{*d}]$.

Definition 7. Une grammaire de tableau à d -dimensions G_A est représentée par :

$$([(N \cup T^{*d}), [T^{*d}], [\mathcal{A}_0], P, \Longrightarrow G_A)$$

où N est l'alphabet des symboles non terminaux, T est l'alphabet des symboles terminaux, $N \cap T = \emptyset$, $\mathcal{A}_0 \in (N \cup T)^{*d}$ est le tableau de départ, P est un ensemble fini de règles de tableau à d dimensions sur V , $V := N \cup T$, et $\Longrightarrow G_A \subseteq [(N \cup T)^{*d}] \times [(N \cup T)^{*d}]$ est la relation de dérivation induite par les règles de tableau dans P .

Une règle de tableau classique à d dimensions p sur V est un triplet $(W, \mathcal{A}_1, \mathcal{A}_2)$ où $W \subseteq \mathbb{Z}_d$ est un ensemble fini, \mathcal{A}_1 et \mathcal{A}_2 sont des applications de W à $V \cup \{\#\}$.

6.3.1 Derivation modes

Dérivation séquentielle : Chaque étape de dérivation exactement une règle est appliquée à un seul objet.

Dérivation maximal (mode t) : On dit que le tableau p dérive un tableau q en mode terminal ou en mode t et on écrit $p \xRightarrow{t} q$ si $p \xRightarrow{*} q$ et $\nexists s$ tel que $q \Longrightarrow s$.

Dérivation en parallèle : Chaque étape de dérivation on applique une règle parmi toutes les règles qui peuvent être appliquée a différents objets (à chaque pas tous les objets qui peuvent évoluer doivent évoluer).

$$\begin{aligned}
p_0 &= \begin{array}{cc} T & T \\ T & \boxed{T} \\ \boxed{T} & \boxed{T} \end{array}, q_0 = \begin{array}{cc} F & F \\ T & \boxed{T} \\ \boxed{T} & \boxed{T} \end{array}, p_1 = \begin{array}{ccc} \boxed{T} & T & \\ \boxed{T} & \boxed{T} & \boxed{T} \end{array}, q_1 = \begin{array}{ccc} \boxed{F} & F & \\ \boxed{T} & \boxed{T} & \boxed{T} \end{array}, \\
p_2 &= \begin{array}{ccc} \boxed{T} & T & T \\ \boxed{T} & \boxed{T} & T \end{array}, q_2 = \begin{array}{ccc} \boxed{F} & F & F \\ \boxed{T} & \boxed{T} & F \end{array}, p_3 = \begin{array}{cc} \boxed{T} & \boxed{T} \\ \boxed{T} & T \\ \boxed{T} & \end{array}, q_3 = \begin{array}{cc} \boxed{T} & \boxed{F} \\ \boxed{T} & F \\ \boxed{T} & \end{array}, \\
p_4 &= \begin{array}{cc} \boxed{T} & \boxed{T} \\ \boxed{T} & T \\ T & T \end{array}, q_4 = \begin{array}{cc} \boxed{T} & \boxed{F} \\ \boxed{T} & F \\ T & F \end{array}, p_5 = \begin{array}{ccc} \boxed{T} & \boxed{T} & \boxed{T} \\ & T & \boxed{T} \end{array}, q_5 = \begin{array}{ccc} \boxed{T} & \boxed{T} & \boxed{T} \\ & F & \boxed{F} \end{array}, \\
p_6 &= \begin{array}{ccc} T & \boxed{T} & \boxed{T} \\ T & T & \boxed{T} \end{array}, q_6 = \begin{array}{ccc} F & \boxed{T} & \boxed{T} \\ F & F & \boxed{F} \end{array}, p_7 = \begin{array}{cc} & \boxed{T} \\ T & \boxed{T} \\ \boxed{T} & \boxed{T} \end{array}, q_7 = \begin{array}{cc} F & \boxed{T} \\ \boxed{F} & \boxed{T} \end{array}, q_8 = \begin{array}{cc} F & \boxed{F} \\ \boxed{F} & \boxed{T} \end{array}.
\end{aligned}$$

Une dérivation dans G_2 s'arrête si et seulement si on finit par appliquer la règle q_8 .

6.4 Conversion aTAM \rightarrow grammaire de tableaux

Theorem 1. Soit $S = (T, \sigma, 1)$ un système aTAM où T est un ensemble de type de tuile. Il existe une grammaire de tableaux $G = (E, A, P, A_0, \Rightarrow_G)$, où E est l'ensemble de notre tuiles, $A \subseteq E$, P est notre ensemble de règles de réécriture et A_0 est notre graine, telle que chaque assemblage de S peut être obtenu dans G avec le mode séquentiel et chaque tableau obtenu dans G peut être obtenu dans S .

6.4.1 Preuve

Démonstration. Soit $t \in T$ une tuile et g_s sa colle sud, pour simuler l'attachement de cette tuile à l'assemblage nous rajouterons à G les règles $\begin{array}{c} \boxed{t} \\ A \end{array}$, pour chaque $A \in T$ dont la colle nord est g_s

Soit $t \in T$ une tuile et g_n sa colle nord, pour simuler l'attachement de cette tuile à l'assemblage nous rajouterons à G les règles $\begin{array}{c} A \\ \boxed{t} \end{array}$, pour chaque $A \in T$ dont la colle sud est g_n

Soit $t \in T$ une tuile et g_e sa colle est, pour simuler l'attachement de cette tuile à l'assemblage nous rajouterons à G les règles $\begin{array}{c} \boxed{t} \\ A \end{array}$, pour chaque $A \in T$ dont la colle ouest est g_e

Soit $t \in T$ une tuile et g_o sa colle ouest, pour simuler l'attachement de cette tuile à l'assemblage nous rajouterons à G les règles $A \rightarrow \boxed{t}$, pour chaque $A \in T$ dont la colle ouest est g_o

Lemma 1. *On prend pas la rotation des tuiles en compte, à la place on peut multiplier la tuile par 4 en changeant les directions Nord, Sud, Est, Ouest*

7 Conclusion

L'incroyable croissance de la nanotechnologie de l'ADN a commencé avec la découverte de Ned Seeman.

Aujourd'hui L'ADN peut faire plus que transporter des informations génétiques, les outils disponibles pour ceux qui travaillent dans ce domaine se sont développés pour inclure une importante bibliothèque de nanostructures à base d'ADN avec des périodicités, des caractéristiques et des structures programmables.

Les principes de conception se sont adaptés pour que les nanostructures d'ADN avec de nouvelles formes arbitraires puissent se réaliser en quelques semaines.

Malgré les progrès significatifs et la puissance de la nanofabrication à base d'ADN, certains problèmes restent à résoudre par exemple on a constaté que les erreurs d'assemblage augmentaient quand le nombre de types de tuiles augmente.

Les stratégies de relecture ou de correction des erreurs pendant et après l'assemblage devront être développées, même si des nanostructures peuvent être assemblés avec des taux d'erreur minimaux, des défis spécifiques associés à chaque application particulière devront être relevés.

Les grammaires des tableaux sont un outils puissant pour représenter l'assemblage des structures basé sur l'ADN plus facilement et nous permettre de prouver des résultats théoriques.

8 Bilan de stage

À l'heure de bilan, je dirais que ce stage a été une expérience très enrichissante tant sur le plan professionnel que personnel. J'ai pu découvrir le domaine de la recherche et de comprendre de manière globale les missions et les difficultés que les chercheurs pouvaient rencontrer. J'ai eu la chance de travailler sur un sujet intéressant qui m'a permis d'avoir une idée sur le monde de la biologie et sur l'utilisation des outils informatique et langages formels dans ce domaine.

8.1 Compétences acquises

1. Communication orale : restitution de travaux en réunion avec d'autres chercheurs qualifiés.
2. Veille documentaire et rédaction des rapports scientifiques et mathématiques en LaTeX.
3. Capacités d'analyse de résultat.
4. Un anglais scientifique et technique.
5. Connaissances en informatique théorique, biologie et bioinformatique.
6. Maîtrise des notions de base en langage de programmation Racket
7. Travail en équipe.

8.2 Difficultés rencontrées et leur résolution

1. Nouveau vocabulaire scientifique (en anglais) et termes difficile à comprendre → Documentation et recherche des définitions.
2. Peu d'informations sur les grammaires de tableaux → Analyse des définitions apparues dans des rapports en anglais et comprendre leurs fonctionnement.

Ce stage m'a beaucoup inspiré et motivé pour continuer à faire de la recherche, je sais que je garderais toujours la possibilité de faire une thèse à l'avenir.