UNIVERSITÉ LOUIS PASTEUR STRASBOURG

Thèse présentée pour obtenir le grade de
Docteur de l'Université Louis Pasteur
Strasbourg I

Discipline : Informatique

par Frey Gabriel

Codes circulaires dans les gènes, évolution et fonction possible

Soutenue publiquement le 2 décembre 2005

Membres du jury

Directeur de Thèse : Michel Christian, Professeur, Université de Strasbourg I
Rapporteur Interne : Tajine Mohamed, Professeur, Université de Strasbourg I
Rapporteur Externe : Bahi Jacques, Professeur, Université de Franche-Comté
Rapporteur Externe : Sagot Marie-France, Directeur de Recherche, Université de Lyon I
Président : Bernot Gilles, Professeur, Université d'Evry

Introduction

La distribution des trinucléotides dans les gènes (séquences codant les protéines) dépend de plusieurs facteurs. En particulier, elle est fonction de l'espèce considérée et du degré d'expressivité du gène. Ainsi, la séquence de trinucléotides d'un gène n'est pas uniquement contrainte par sa traduction en acides aminés. Le biais dans l'usage des codons synonymes en est la meilleure illustration. Afin d'identifier les occurrences de motifs particuliers, la distribution des trinucléotides dans les phases décalées a été étudiée.

Trois phases de lecture potentielles des gènes sont considérées : la phase 0 qui commence par un codon d'initiation et qui correspond au cadre de lecture d'un gène, et les phases 0 et 1, qui sont la phase 0 décalée respectivement de 1 et 2 nucléotides dans le sens 5'-3' (vers la droite). Les fréquences d'occurrence des 64 trinucléotides sont calculées dans chacune des 3 phases (64*3 valeurs). Ainsi, la fréquence la plus élevée dans l'une des 3 phases permet d'assigner un trinucléotide à une phase préférentielle. Trois ensembles $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$ peuvent être identifiés. Ils correspondent à l'ensemble des trinucléotides dont la phase préférentielle est respectivement la phase 0, 1 et 2. Une étude ancienne a montré que les 3 ensembles de 20 mots $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$ (sans les 4 trinucléotides formés de lettres identiques) identifiés dans des gènes de procaryotes et eucaryotes, forment des codes circulaires associant un code circulaire à chacune des 3 phases potentielles des séquences biologiques et appelés C^3.

Un code circulaire est une classe particulière de codes. C'est un ensemble de mots tel que, tout mot écrit sur un cercle (la lettre qui suit la dernière lettre du mot est la première lettre du mot) a, au plus, une décomposition unique en mots du code. Le décodage d'un mot écrit à partir de mots d'un code circulaire est donc non ambigu. Il est possible de le retrouver après la lecture d'une suite de quelques nucléotides (appelée fenêtre du code) n'importe où dans le mot. On peut montrer de façon évidente, qu'un code circulaire ne peut pas contenir de trinucléotides AAA, CCC, GGG ou TTT, ni de trinucléotides permutés (trinucléotides équivalents par permutation circulaire, comme par exemple AAC et ACA).

Fonction biologique des codes circulaires

Une fonction biologique possible des codes circulaires serait la génération de gènes dont le cadre de lecture pourrait être déterminé automatiquement et localement, c'est-à-dire sans décodage à partir du codon d'initiation. De plus, les mots composés à partir de mots d'un code circulaire peuvent aussi servir de marqueurs pour des régions particulières du

gène, permettant ainsi de les localiser et surtout de recadrer rapidement leur lecture. L'information contenue dans les codes circulaires associés aux génomes permet de recadrer leur phase de lecture. Une nouvelle méthode de factorisation a été développée afin de retrouver le cadre de décodage des séquences réelles (donc non composées uniquement de mots de codes circulaires) à partir des codes circulaires. Les résultats obtenus montrent qu'il est possible de retrouver la phase de lecture, même à partir de mots courts (inférieurs à 25 lettres), et que les mots de codes circulaires sont plus performants que les mots de fréquences maximales.

Recherche massive de codes circulaires dans les génomes

L'existence de biais dans l'usage des codons synonymes et de codes génétiques variants, nous a suggéré que les gènes contenus dans des génomes différents pourraient être associés à des codes circulaires différents et non au seul code identifié en 1996. La première partie de cette thèse s'est intéressée à ce problème avec une recherche statistique massive de codes circulaires sur tous les génomes complets connus lors de ce travail. Cette approche a nécessité le développement d'une nouvelle méthode statistique quantitative, automatique et sensible, nommée FPTF (Frame Permutated Trinucleotide Frequency), pour assigner un trinucléotide à une phase. Elle considère ainsi les 20 groupes de 3 nucléotides permutés. Une phase préférentielle différente est assignée à chaque trinucléotide du groupe. Une valeur numérique est attribuée aux assignations trinucléotide/phase préférentielle, permettant ainsi de comparer au cas aléatoire ayant une valeur de 1/3. Cette méthode permet ainsi de rechercher les codes C^3 (codes circulaires composés de 20 mots, c'est-à-dire maximaux, et dont les ensembles obtenus par permutation circulaire de chacun des mots forment aussi des codes circulaires).

Cette méthode a été appliquée à 191 génomes complets : 16 génomes d'archaea et 175 génomes de bactéries. Elle a permis d'identifier de nouveaux codes circulaires C^3 : 15 pour les archaea et 72 pour les bactéries, confirmant ainsi notre hypothèse initiale sur l'existence de plusieurs codes circulaires dans les diverses familles de gènes.

Analyse des codes circulaires identifiés

La probabilité d'obtenir de tels codes C^3 est très faible. En effet, un ensemble de 20 trinucléotides non identiques et non permutés a une probabilité seulement de 3.7×10^{-3} d'être un code C^3. Diverses propriétés de ces nouveaux codes ont également été étudiées. En particulier, le calcul de la phase préférentielle sur l'alphabet R,Y montre que le motif

ancestral RNY (R=A,G, Y=C,T, N=A,C,G,T) est présent dans la majorité des 87 codes circulaires. Ces divers codes permettent de coder entre 9 et 15 acides aminés. Leurs fenêtres pour retrouver les phases de lecture varient entre 5 et 13 nucléotides.

Modèle d'évolution des codes circulaires

Un modèle analytique d'évolution basé sur une matrice de trinucléotides 64*64 à 6 paramètres associés aux transitions et transversions aux 3 positions possibles des trinucléotides) a été développé. Il généralise tous les modèles actuels basés sur des matrices de mutation 4*4 à plusieurs paramètres et 64*64 à 3 paramètres. Il permet de déterminer à un instant t les probabilités exactes d'occurrence de chaque trinucléotide muté en fonction des 6 paramètres de substitutions. La recherche des valeurs propres et des vecteurs propres du système différentiel ne peut pas être obtenue directement avec les logiciels de calcul formel (Mathematica, Matlab). Elle a été réalisée en utilisant des propriétés des sous-matrices blocs de la matrice de mutation. Les manipulations algébriques par calcul formel permettent d'obtenir des formules analytiques comportant plusieurs dizaines de termes exponentiels qui peuvent être mis en fonction des valeurs propres.

Une application de ce modèle permet de montrer que les codes actuels d'archaea ont pu dériver par mutation du code commun $\mathcal{X}_0$ pour certaines valeurs particulières des paramètres de substitutions.

Table des matières

Chapitre 1

Codes dans les gènes

La transmission des caractéristiques d'un individu à ces descendants est un phénomène observé de longue date. La persistance héréditaire de certains traits phénotypiques permet raisonnablement de supposer l'existence d'une information génétique transmissible au cours des générations. Il a fallu attendre le milieu du XX^{eme} siècle pour commencer à appréhender la machinerie moléculaire à l'oeuvre lors du stockage cellulaire de l'information génétique.

En 1953, grâce aux travaux de James Watson et Francis Crick [WC53], la structure de l'ADN est identifiée. Il s'agit d'une succession de paires de bases nucléiques (A-T, C-G) formant une double hélice (figure 1.1).

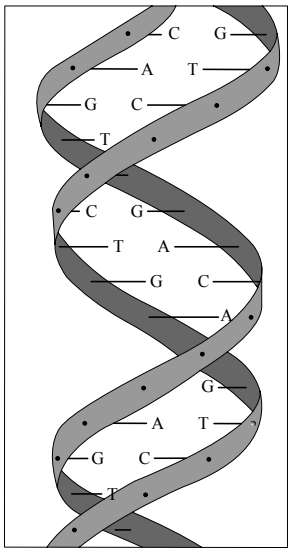

FIG. 1.1 – Structure de l'ADN

Dans ce chapitre, différents aspects du codage de l'information génétique sont présentés. Nous commençons par décrire les différentes étapes permettant de synthétiser une protéine à partir de l'information stockée dans la séquence d'ADN. Le code génétique, son historique, sa structure ainsi que ses possibilités de variations sont discutés. Les possibilités de superposition de messages encodés par des codes biologiques (codes génétiques ou autres) seront discutées. Le maintien de la phase de lecture des gènes étant un mécanisme complexe et encore non totalement compris, des erreurs peuvent se produire, provoquant une lecture non conventionnelle des gènes.

Dans le message formé par la séquence de bases nucléiques est encodé le protéome complet de l'individu. Différents mécanismes sont à l'oeuvre lors de la lecture de la séquence d'ADN.

1.1 Etapes de la synthèse des protéines

Dans ce travail, il sera fait référence aux différentes étapes de la synthèse des protéines et aux diverses molécules impliquées dans ces processus. Un bref rappel est donc donné dans ce chapitre. Les processus présentés sont préférentiellement orientés vers les modèles bactériens.

De l'ADN à l'ARN messager : la transcription

Les protéines ne sont pas obtenues directement à partir de l'ADN. Il existe plusieurs étapes intermédiaires. L'ADN est d'abord utilisé comme modèle pour la synthèse d'une molécule simple brin similaire appelé acide ribonucléique (ARN). L'ARN, comme l'ADN, est constitué de bases chimiques attachées séquentiellement. Lors de la transcription de l'ADN, la molécule d'ARN produite est strictement identique à l'exception des bases nucléiques uraciles (U) qui remplacent les bases thymines (T) de l'ADN. La transcription se passe de la façon suivante. Lors du cycle cellulaire, une enzyme, l'ARN polymérase, s'attache à la molécule d'ADN et sépare les deux brins (figure 1.2). Un des brins est alors utilisé comme modèle sur lequel les bases complémentaires viennent s'apparier (A est apparié à U).

Plusieurs types d'ARN sont obtenus par transcription. La majorité des ARN sont utilisés comme intermédiaires dans la production de protéines à partir de gènes et sont appelés ARN messager (ARNm). Il existe d'autres types d'ARN qui seront présentés dans les sections suivantes. Dans le cas des eucaryotes, les molécules d'ARN doivent migrer du noyau vers le cytoplasme pour que la synthèse de protéine puisse être réalisée.

De l'ARNm au protéine : la traduction

L'ARNm est utilisé à son tour comme modèle pour synthétiser les protéines lors d'une étape appelée traduction. Les protéines sont des chaînes d'acides aminés de différentes longueurs. Le type des acides aminés de la chaîne est déterminé à partir de l'ARNm. Les principaux composants biologiques impliqués dans ce processus sont l'ARNm, les ribosomes et les ARNt (figure 1.3).

Les ribosomes peuvent être décrits comme étant des molécules sphérique qui parcourent l'ARNm et qui mettent en correspondance les séquences de nucléotides et les acides aminés. Les ribosomes sont des assemblages complexes, constitués d'un tiers de protéines et de deux tiers d'ARN ribosomique (catégories particulières d'ARN).

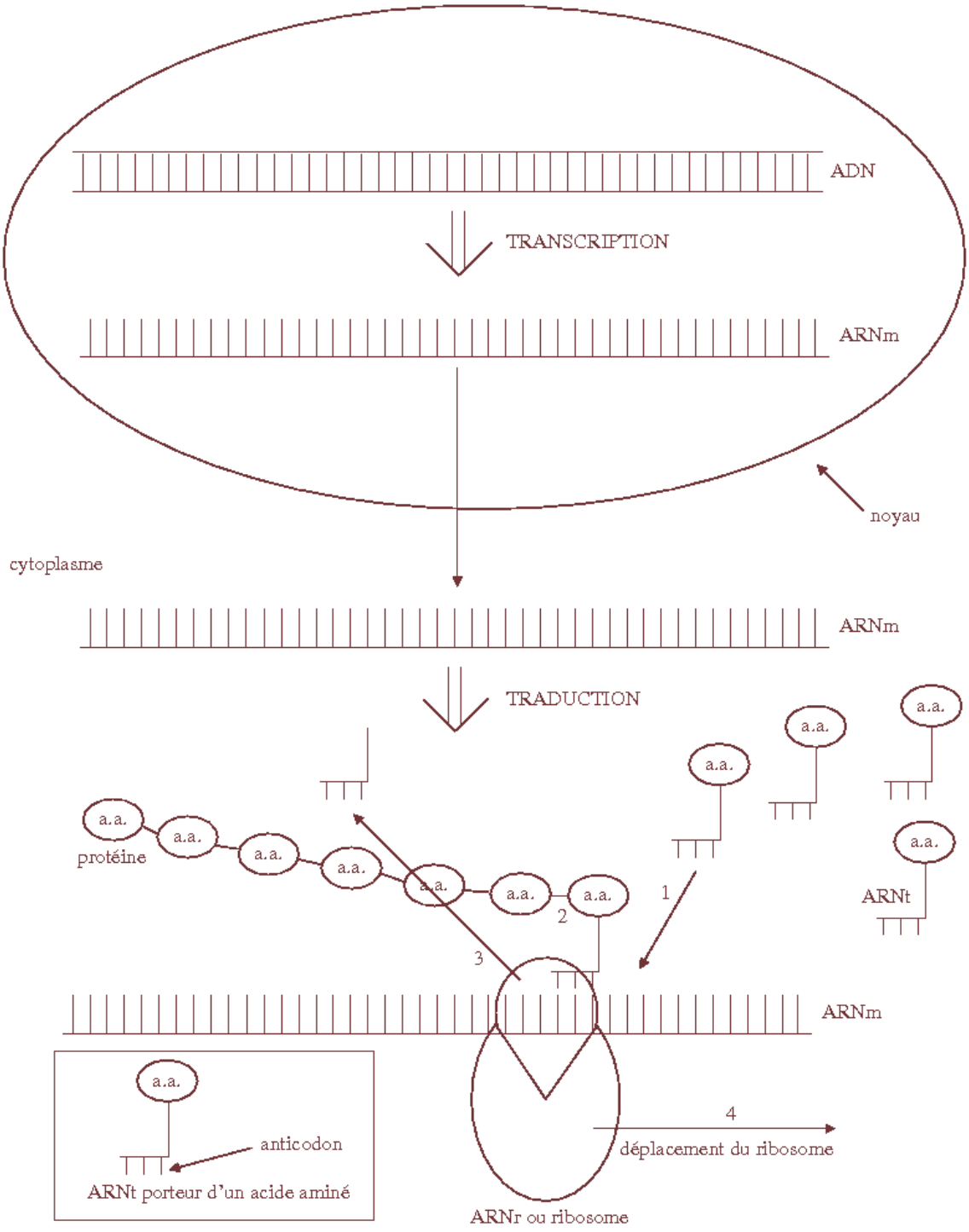

FIG. 1.2 – Transcription de l'ADN en ARNm et traduction de l'ARNm en chaînes d'acides aminés.

Les ARN de transfert (ARNt) associent un trinucléotide particulier à un acide aminé. Chaque ARNt porte un acide aminé. Le choix de l'acide aminé est décidé par la lecture de trois bases nucléiques de l'ARNm. Un ARNt assure cette fonction en associant un acide aminé se trouvant à l'une de ses extrémités à trois de ces bases particulières nommées anti-codon, capable de s'apparier sélectivement à certains triplets de l'ARNm. Cette appariement se fait selon les règles standard de l'appariement nucléique (A avec U et G avec C) pour deux des bases, la troisième obéissant à une règle particulière (règles du wobble, voir 1.3.2).

Les ARNt associés à leurs acides aminés sont diffusés au ribosome. La position du ribosome sur l'ARNm détermine le trinucléotide actuellement lu. L'ARNt correspondant s'apparie à ce triplet, l'acide aminé porté par 'ARNt est alors ajouté à la chaîne croissante d'acides aminés formant la protéine. Le ribosome avance alors de trois bases pour lire le trinucléotide suivant. La séquence d'ARNm peut être lu simultanément par plusieurs ribosomes situés à différents endroits de la séquence.

La chaîne d'acides aminés est libérée lorsqu'un trinucléotide particulier appelé codon stop est lu (TAT, TAA et TGA dans le code universel). Aucun acide aminé n'est associé à ces trinucléotides mais ils sont reconnus par des protéines connues sous le nom de "release factor" qui détachent les ribosomes de l'ARNm et provoquent la libération de la chaîne d'acide aminée.

Il faut signaler que les mécanismes exactes permettant au ribosome de décider du début ou de la fin de la traduction ainsi que le mécanisme permettant de maintenir la lecture des trinucléotides en phases ne sont pas encore parfaitement connus (voir erreur de lecture et recodage en section 1.5).

Pour que la traduction soit fiable, il est impératif qu'à un ARNt spécifique soit toujours associé un même acide aminé. Cette association est effectuée par une enzyme appelée aminoacyl-ARNt synthétase, dont la fonction est de reconnaître un ARNt et d'y attacher l'acide aminé correspondant.

Par conséquent, l'association d'un acide aminé à un codon dépend de l'ARNt et de l'aminoacyl-ARNt synthétase. Ces deux structures ont leur structure spécifiée par l'ADN des organismes : l'ARNt est obtenu par transcription, les aminoacyl-ARNt synthétases sont des protéines et sont donc obtenues par transcription et traduction. Ces molécules

sont donc susceptibles d'être altérées par mutation. Les associations trinucléotides/acide aminé et par conséquent le code génétique peuvent être modifiés.

1.2 Découverte du code génétique

Son décodage permet, à partir d'une suite de bases nucléiques (alphabet de 4 lettres), d'obtenir les séquences d'acides aminés (alphabet de 20 lettres) formant les protéines. Les moyens techniques de l'époque ne permettaient pas de connaître de façon expérimentale le code utilisé par la nature. Diverses propositions de codes ont alors été faites (un historique est présenté dans [Hay98]), dont celle de George Gamow [Gam54] avec le *Diamond Code*.

1.2.1 Le Diamond Code

Gamow émit l'hypothèse que la double hélice d'ADN formait une paire de rails à l'intérieur desquels se placent les acides aminés, chaque acide aminé se positionne entre les 2 bases d'une même paire. L'hypothèse était intéressante car l'espacement entre les bases était le même que l'espacement entre les acides aminés. L'acide aminé capable de se placer dépendait de la forme du site créé par les 4 bases : les 2 bases de la paire en question, une base de la paire "précédente" et une base de la paire "suivante" (voir Fig 1.3).

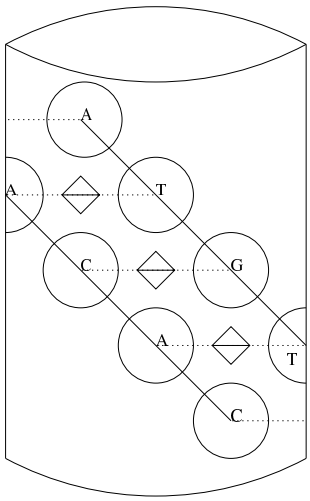

FIG. 1.3 – Placement des acides aminés selon le diamond code

La paire de bases obéissant à la règle de complémentarité, pouvait prendre deux valeurs A-T (T-A) ou C-G (G-C). De plus, sous l'hypothèse que l'acide aminé était le même si l'on échangeait les places des bases "précédente" et "suivante", 20 combinaisons sont possibles (voir Tab 1.1), et correspondent exactement au nombre d'acides aminés.

Le *Diamond code* fût le premier code à être envisagé comme étant une solution possible au problème du décodage de l'information génétique contenue dans l'ADN. Un de ces avantages est que la lecture se fait avec un décalage d'une base à la fois, hormis aux

A A⇜⇝ T A *01*	A A⇜⇝ T C *02*	A A⇜⇝ T G *03*	A A⇜⇝ T T *04*	C A⇜⇝ T C *05*
C A⇜⇝ T G *06*	C A⇜⇝ T T *07*	G A⇜⇝ T G *08*	G A⇜⇝ T T *09*	T A⇜⇝ T T *10*
A C⇜⇝ G A *11*	A C⇜⇝ G C *12*	A C⇜⇝ G G *13*	A C⇜⇝ G T *14*	C C⇜⇝ G C *15*
C C⇜⇝ G G *16*	C C⇜⇝ G T *17*	G C⇜⇝ G G *18*	G C⇜⇝ G T *19*	T C⇜⇝ G T *20*

TAB. 1.1 – Le diamond code

extrémités de l'hélice. Chaque base appartient donc simultanément à 3 triplets. Il n'y a donc pas de phase de lecture à identifier. Ce type de code est dit *chevauchant*. Ce code était donc très efficace puisque le ratio bases/acides aminés tend vers 1 quand le nombre de bases tend vers l'infini.

Toutefois, s'il peut coder chaque acide aminé, ce code n'est pas capable de coder chaque suite de 2 acides aminés (dipeptides). En effet, pour 20 acides aminés, il existe $20^2 = 400$ dipeptides. Sur le diamond code, un dipeptide est codé par 4 paramètres (une base sur le premier brin d'ADN, 2 paires de bases appariées et une seconde base sur le second brin). Le *diamond code* ne peut donc coder que pour $4^4 = 256$ dipeptides. Il ne peut donc générer qu'une variété réduite de séquences d'acides aminés. Il fût définitivement abandonné lorsque Crick identifia expérimentalement dans les protéines des suites d'acides aminés ne pouvant être générées par le *D*iamond Code.

1.2.2 Les codes comma free

Gamow proposa encore deux codes ([Gam54], [Gam56]) ayant la propriété de *chevauchement*. Le fait que le chevauchement soit utilisé dans la nature fut remis en question, notamment par le fait que la mutation d'une base peut provoquer la modification simultanée de 3 acides aminés. Les travaux de Sydney Brenner [Bre57] ont définitivement exclu la possibilité d'un code chevauchant. En effet, Brenner a fait décoder une séquence avant et

après avoir muté un unique nucléotide. Comme les 2 protéines synthétisées ne différaient que d'un acide aminé, le code génétique ne pouvait pas avoir la propriété de chevauchement.

Sans chevauchement, se pose le problème de la phase de lecture. En partant du principe que chaque acide aminé est codé par un trinucléotide, il y a trois phases de lecture possibles, et une seule phase est valide. Pour éviter se problème, Crick soumit en 1957 ([CG57]) l'hypothèse que seul 20 triplets de nucléotides sont valides et codent les 20 acides aminés. Les autres trinucléotides ne sont pas porteurs d'information et ne peuvent apparaître que dans les phases décalées ou dans les zones non-codantes. Il fallait donc trouver un code tel que :

- Il contienne 20 triplets (ou codons).
- En plaçant 2 codons quelconques l'un derrière l'autre, les codons lisibles en phases décalées n'appartiennent pas au code.

Un code offrant ces propriétés est dit *comma free* ou *comma-less* car un texte écrit sur un tel code, sans séparer les différents mots, est parfaitement lisible (la définition formelle des codes circulaires se trouve dans 3.6).

James Crick, John Griffith et Leslie Orgel ont montré [CGL57] que les codons AAA, CCC, GGG et TTT/UUU ne peuvent appartenir à ce type de code. En effet, les textes composés uniquement de lettres identiques sont lisibles dans plusieurs phases. Il reste donc 60 codons. Un mot et son permuté circulaire (une partie du début du mot devient la fin du mot) ne peuvent être tous les deux simultanément dans un même code comma free. Par exemple, avec le code {CGA, GAC} et le texte "CGACGA", GAC peut être lu en phase 1 bien qu'appartenant déjà au code. Les 60 codons sont donc divisés en groupe de 3 codons. Chaque groupe contient un codon et ses 2 permutés (ex : ACG, CGA et GAC). Il y a donc 20 groupes. Ce nombre correspond exactement au nombre d'acides aminés. Crick, Griffith et Orgel réussirent à identifier plusieurs codes comma free de 20 mots. Les codes comma free présentent cependant de fortes contraintes. En particulier, la plupart des mutations de codons génèrent un codon qui n'appartient plus au code, provoquant ainsi l'arrêt de la lecture de la séquence. Les codes comma free sont donc très peu tolérant aux mutations.

De nombreux théoriciens se sont intéressés à cette classe de code. En particulier, Solomon W. Golomb identifia les 408 codes possibles de 20 mots de 3 lettres sur un alphabet de 4 lettres [GWD58]. En considérant les contraintes des alphabets biologiques, il développa d'autres classes de codes particulières, comme les codes circulaires.

Pendant 5 ans, la communauté scientifique est persuadée que le code génétique est un code comma free. En 1961, Marshall W. Nirenberg et J. Heinrich Matthaei [NM61] arrivent à synthétiser une chaîne de U (uracile) et la chaîne d'acides aminés associée. Dans les codes comma free, le codon UUU n'a aucun sens car il est impossible de trouver la phase de lecture dans une suite de lettres identiques. Pourtant après décodage, Nirenberg et Matthaei montrent que le triplet UUU code pour un acide aminé : la phenylanaline. En 1965, la totalité du code génétique est trouvée de façon expérimentale. Il ne correspond à aucun des codes théoriques proposés.

1.3 Le code génétique

Le code génétique est très différent des codes comma free, diamond codes et autres codes proposés durant les années 50-60 (table 1.2).

Codon	**A. A.**	**Codon**	**A. A.**	**Codon**	**A. A.**	**Codon**	**A. A.**
AAA	Lys	CAA	Gln	GAA	Glu	TAA	stop
AAC	Asn	CAC	His	GAC	Asp	TAC	Tyr
AAG	Lys	CAG	Gln	GAG	Glu	TAG	stop
AAT	Asn	CAT	His	GAT	Asp	TAT	Tyr
ACA	Thr	CCA	Pro	GCA	Ala	TCA	Ser
ACC	Thr	CCC	Pro	GCC	Ala	TCC	Ser
ACG	Thr	CCG	Pro	GCG	Ala	TCG	Ser
ACT	Thr	CCT	Pro	GCT	Ala	TCT	Ser
AGA	Arg	CGA	Arg	GGA	Gly	TGA	stop
AGC	Ser	CGC	Arg	GGC	Gly	TGC	Trp
AGG	Arg	CGG	Arg	GGG	Gly	TGG	Cys
AGT	Ser	CGT	Arg	GGT	Gly	TGT	Cys
ATA	Ile	CTA	Leu	GTA	Val	TTA	Leu
ATC	Ile	CTC	Leu	GTC	Val	TTC	Phe
ATG	Met	CTG	Leu	GTG	Val	TTG	Leu
ATT	Ile	CTT	Leu	GTT	Val	TTT	Phe

TAB. 1.2 – Le code génétique standard

Nom	Abréviation	Lettre
Alanine	Ala	A
Arginine	Arg	R
Asparagine	Asn	N
Acide Aspartique	Asp	D
Cystéine	Cys	C
Glutamine	Gln	Q
Acide Glutamique	Glu	E
Glycine	Gly	G
Histidine	His	H
Isoleucine	Ile	I
Leucine	Leu	L
Lysine	Lys	K
Méthionine	Met	M
Phénylalanine	Phe	F
Proline	Pro	P
Sérine	Ser	S
Thréonine	Thr	T
Tryptophane	Trp	W
Thyrosine	Tyr	Y
Valine	Val	V

TAB. 1.3 – Table des acides aminés

1.3.1 Redondance et neutralité du code génétique

20 acides aminés (table 1.3) sont associés aux 64 trinucléotides (table 1.2). Plusieurs trinucléotides ont aussi des rôles de "balises" (début et fin de traduction). Par conséquent, si tous les trinucléotides ont une fonction de codage en acide aminé alors il existe nécessairement une redondance et plusieurs codons doivent être associés au même acide aminé. On parle de dégénérescence du code génétique.

Le changement ayant la plus faible conséquence sur un message biologique est la substitution dans une séquence d'une base par une autre. Les mutations sont suffisamment rares pour rendre négligeable la probabilité que deux bases d'un même triplet soient modifiées en même temps. Deux trinucléotides ne variant que d'un nucléotide sont considérés comme

étant voisin. Une mutation d'un nucléotide d'un codon est dit neutre si l'acide aminé associé au codon avant et après la mutation est le même. Le code génétique possède certaines propriétés par rapport aux substitutions simples.

Ainsi, les mutations sur la première base sont très rarement neutres (cas pour certains codons associés à la leucine ou à l'arginine), par exemple AGG et CGG codent tous les deux pour la leucine. Les mutations de la seconde base sont uniquement neutres pour les codons stop UAA et UGA. Les substitutions sur la troisième base sont souvent neutres. Dès lors, les triplets synonymes ne sont pas répartis aléatoirement sur la table du code génétique mais appartiennent fréquemment à un même bloc correspondant aux trinucléotides dont les deux premiers nucléotides identiques mais le troisième est variant. Par exemple, tous les codons commençant par CG codent pour la proline. La répartition des codons dans la table du code génétique peut être expliqué en partie grâce aux règles du wobble.

1.3.2 Règles du wobble

Chaque trinucléotide n'est pas nécessairement associé à un ARNt différent. Un même ARNt peut dans certains cas reconnaître plus d'un codon. Par exemple, l'ARNt associé à l'alanine peut s'associer aux trinucléotides GCU, GCC et GCA et de façon plus rare à GCG. Cette observation ainsi que d'autre a permis à Crick ([Cri66]) de proposer les règles du wobble : l'appariement codon-anticodon sur les deux premières positions obéit aux règles classiques d'appariement tandis que l'appariement sur la troisième position est moins discriminant et respecte des règles particulières (voir table 1.4). Les raisons de cette liaison plus fragile sur la troisième base peuvent être expliquées par des raisons moléculaires [Plu94].

Première base de l'anticodon	**Dernière base du codon**
A	U
C	G
G	U ou C
U	A ou G
I	A, U ou C

TAB. 1.4 – Règle du wobble

Les règles du wobble disent que seuls A et C obéissent à la règle traditionnelle d'appariement tandis que G, C et I (l'inosine base trouvée généralement dans l'ARNt) sont moins

discriminantes et font que les ARNt peuvent accepter un certain nombre de trinucléotides. En particulier, l'inosine à la première position de l'anticodon peut être appariée à A, U et C à la troisième position du trinucléotide. Des ARNt avec une inosine à la première position de la l'anticodon apparaissent pour chacun des 8 blocs de codons synonymes. Ils ne concernent que 3 des 4 codons d'un bloc et par conséquent un autre ARNt avec un C à la première base de l'anticodon est nécessaire (G pour la 3^{eme} base du codon). Les règles du wobble concernent aussi le cas commun dans le code génétique où pour les deux premières lettres identiques, les triplets terminant par U et C ont un sens tandis que les triplets se finissant par A et G en ont un autre.

Le code génétique a longtemps été considéré comme universel et invariant. Il est désormais connu qu'il peut lui aussi subir des modifications au cours de l'évolution.

Evolution du code génétique

Plusieurs propositions ont été faites sur l'origine du code génétique et ses possibilités d'évolution. Pendant longtemps, la théorie la plus communément admise était celle du "frozen accident" proposée par Francis Crick [Cri68]. Elle considère que le code génétique est figé et universel. Les réassignements de codon à des acides aminés différents n'étaient possibles que dans les étapes primitives de la vie, lorsque le fonctionnement des cellules reposait sur un petit nombre de protéines. Le code génétique se serait stabilisé à un moment de l'évolution. Le choix des assignations codon / acide aminé aurait été en partie arbitraire (accident de l'évolution). Tout changement du code génétique des organismes modernes altérerait la signification des codons, ce qui entraînerait une modification de tous les gènes et par conséquent de tout le protéome. Tout message traduit serait donc susceptible d'avoir des erreurs délétères.

Cette théorie était en opposition à l'hypothèse stéréochimique [Woe65]. Cette dernière suppose que la relation entre l'anticodon de l'ARNt et l'acide aminé qu'il transporte est due à une affinité d'une nature inconnue entre le codon et l'acide aminé. Cette affinité serait à l'origine du code à une époque où les ARNt n'était pas encore disponibles pour assurer la traduction. Cette hypothèse présente deux avantages. Premièrement, elle propose une explication sur la façon dont le code pourrait avoir été défini même en l'absence de ARNt. Ensuite, l'universalité du code serait une conséquence logique de la structuration chimique de l'assignation entre codons et acides aminés.

La théorie d'un code universel a du être remise en question lorsque des codes variants ont été identifiés pour plusieurs espèces [BDD79]. Aucun des codes variants actuellement observés n'est très différent du code universel et tous codent pour le même nombre d'acides aminés. Certains codons sont plus susceptibles d'être modifiés que d'autres, les significations alternatives étant toujours les mêmes. Le code identifié dans les mitochondries de levure est le plus différent avec 6 codons ayant une signification différente, les codes des mitochondries étant généralement plus variant que ceux du noyau (voir [KFL01] pour une liste détaillé des codes variants). Ces codes sont généralement considérés comme étant exceptionnels. Cependant un grand nombre de codes pourrait être identifié par l'examen des espèces non encore étudiés.

Le code génétique a donc évolué. Il est dès lors raisonnable de supposer qu'il a subit une certaine forme d'optimisation. Il a été observé que pour le code génétique, les acides aminés codés par des trinucléotides proches les uns des autres ont des propriétés chimiques similaire, les mutations sont donc globalement peu pénalisantes pour la protéine. Les acides aminés modifiés sont remplacés par des acides aminés pouvant jouer un rôle similaire pour la structure de la protéine afin que la fonction soit conservée. Pour ce critère d'optimisation aux mutations, Le code génétique est meilleur pour ce critère à la plupart des codes possibles ([FH98], [Fre00]).

La sélection d'un code génétique optimale par l'évolution est cependant complexe. Généralement, les mutations ponctuelles ont des avantages ou désavantages graduelles et sont soumis à la sélection par l'évolution. L'évolution du code génétique peut être décrite comme une méta-évolution. Ce n'est pas une simple propriété de l'individu qui est modifiée mais la relation d'un individu avec ses organismes similaires. Tout le codage du protéome est donc modifié.

Des étapes intermédiaires ont été envisagées pour expliquer les variations du code génétique. Dans l'hypothèse de capture de codon, pour une espèce, l'utilisation d'un codon particulier diminue jusqu'à devenir pratiquement nulle. Dès lors, le réassignement de ce codon à un autre acide aminé aurait des conséquences plus restreintes sur les gènes. Une autre proposition suggère que les ARNt de transferts peuvent être par moment d'avantage ambigus (règles du wobble étendues sur d'autre positions, autres bases methylés sur l'anticodon, etc.). Différents acides aminés peuvent alors être associés à un même codon pendant une période de l'évolution d'une espèce.

Le code génétique est le signal le plus présent dans les gènes. Cependant, d'autres codes associés à des signaux biologiques peuvent apparaître dans les séquences nucléiques.

1.4 Codes chevauchants

Les génomes sont composés de contraintes spécifiques superposées, qui se réfèrent à des phénomènes biologiques distincts [Tri89]. De façon générale, tous les motifs de séquences associés à une certaine fonction biologique peuvent être considérés comme un code. Le code génétique est donc un code particulier. Divers signaux servent à la régulation de la traduction [Roc99] et au contrôle de la transcription (promoteurs, opérateurs, terminateurs ou anti-terminateurs) [Plat98].

En conséquence, dans le texte génomique se superposent plusieurs contraintes différentes. Toutes ces contraintes ne sont pas forcément constituées de mots de longueurs et séquences précises. Les mots peuvent être dans certain uniquement caractérisé par diverses propriétés (composition à partir de mots élémentaires, biais particulier dans l'utilisation de nucléotides, etc...). La coexistence simultanée de diverses contraintes dans un même espace de codage, l'ADN, implique que celles-ci ne sont pas indépendantes entre elles. Le bon fonctionnement des organismes nécessite que la cohabitation de ces contraintes soit la moins conflictuelle possible.

L'identification de ces contraintes passe par la caractérisation des signaux associés aux différents processus (traduction, transcription, etc). Pour chaque région, des contraintes spécifiques peuvent interagir entre elles. La signification des mots est alors fortement contextuelle. Un exemple est la séquence de Shine-Dalgarno qui, bien que présente régulièrement dans les gènes a une signification d'indication de site de fixation du ribosome qu'au début des gènes lorsqu'elle est à proximité d'un codon de début de traduction [VR92]. La méthode de recherche doit aussi prendre en compte que des contraintes puissent être caractérisées par des mots fortement différents, soit par leur taille, soit par le fait que quelques-uns sont exacts et que d'autres correspondent à des variations de motifs par rapport à un consensus. Certains motifs peuvent aussi être caractérisés par des informations d'un autre ordre comme les structures secondaires de l'ARN encodant le message.

Les séquences d'ADN sont souvent comparées aux textes écrits en langage naturel. Une différence notable est donc la possibilité de superposition de messages pour les gènes. La possibilité de message chevauchant a donc été prédite dès 1968 par Holliday (Hol68) qui considéré que les signaux caractérisant la recombinaison devaient être contenus dans les séquences codant les protéines. La superposition de message est possible grâce à la dégénérescence du code génétique et à la variabilité des séquences d'acides aminés. En

effet une protéine possède un certain nombre de positions pour lesquelles certains acides aminés sont nécessaires tandis que d'autres positions sont moins contraintes et peuvent donc être fortement variables selon leurs acides aminés associés. Comme exemple de messages chevauchants, il est possible de citer les structures secondaires de l'ARN apparaissant dans des zones codantes et pouvant correspondre à des rôles fonctionnels. Certaines périodicités de l'ARNm sont aussi supposées avoir un rôle dans le recodage (voir 1.5.2).

1.4.1 Identification de messages chevauchants

Les messages se superposant au code génétique peuvent être identifiés sans connaissance a priori sur leur signification. L'utilisation de méthodes statistiques appliquées à des séquences liées suivant certains critères permet de caractériser les signaux apparaissant de façon concordante. Il est cependant indispensable de distinguer dans les séquences codantes parmi les signaux identifiés, ceux liés à des propriétés fonctionnelles de la protéine de ceux correspondant à des signaux de régulation de la gestion de l'information dans les séquences.

1.4.2 Sélection des messages chevauchant

Un certain nombre d'erreurs peuvent se produire durant la réplication. Ces mutations sont une source de nouveauté dans les séquences, permettant ainsi d'accroître le nombre de versions d'une séquence et même de proposer de nouveaux modèles de protéines à partir de modifications de structures existantes. Mais les mutations peuvent aussi être néfastes à un organisme en rendant certaines des séquences non fonctionnelles. Une solution pour protéger les gènes de mutations délétères est de proposer de multiples copies d'un même gène. Cependant, la redondance n'est pas possible pour tous l es organismes.

En effet, les espèces disposant d'une faible qualité de réplication ne peuvent maintenir que de courts génomes qui peuvent se révéler insuffisant pour contenir de façon séquentielle toutes les informations biologiques nécessaires à la survie de l'organisme. Pour accroître la quantité d'information pouvant être stockée, il devient nécessaire d'augmenter le nombre d'informations par unité de longueur. Il a alors été proposé que certaines parties des génomes pourraient coder simultanément plusieurs informations [HKH93]. Dans le cas de nombreuses mutations, la superposition de messages cruciaux diminue le nombre de zones à conserver. De plus, les modifications de séquences codant plusieurs fonctions auront des effets plus importants sur l'organisme, ce qui peut favoriser leur sélection et leur stabilité au cours de l'évolution ([Kon92], [WS99]). Cependant les messages superposés contraignent fortement la séquence et en diminuent donc l'adaptabilité pour des modifications mineures

[HH92]. Les séquences correspondant à des zones de messages chevauchant utilisent des codons avec un fort taux de dégénérescence. Cette différence d'utilisation de codons est même une caractéristique des zones de gènes chevauchants [Pav97].

Un modèle mathématique a été proposé [Kra00] pour évaluer la stabilité des messages chevauchant au cours de l'évolution. Le coût informationnel des superpositions est alors estimé. Le modèle montre que le chevauchement favorise le couplage à la traduction de messages liés fonctionnellement (voir aussi [Ino00]). Une modification d'un message a des conséquences sur l'autre message. La superposition de messages qui serait déjà effectivement liés serait dès lors moins contraignante. Les messages chevauchants pourraient provenir de la réduction puis de la suppression de zones intergéniques par des biais mutationnels [Cla01].

La pénalité informationnelle associée à la superposition des messages a été estimée de plusieurs façons. Elle consiste de façon générale à évaluer la probabilité qu'une modification d'un message a des conséquences délétères sur l'autre message. La survivabilité et la qualité d'adaptation de séquences proposant des chevauchements dépens alors du rapport entre le taux de mutation et la pénalité informationnelle induit par la superposition des messages. Ainsi un taux de mutation faible et une incompatibilité entre les séquences rend le chevauchement peux judicieux tandis qu'un fort taux de mutation et une bonne superposabilité des séquences rend leur coexistence simultanée positive pour l'organisme.

1.5 Modification de la phase de lecture

1.5.1 Les erreurs de traduction

Initialement, le monde vivant aurait été majoritairement composé d'ARN (RNA world décrit dans [Sha99]). Puis une composition mixte d'acides nucléiques et de protéines serait apparue. Cette transition a nécessité l'apparition d'une machine moléculaire, le ribosome. Ainsi, les processus biologiques initialement gérés par des ribozymes (enzymes ARN) ont pu être effectués par des protéines, l'ARN conservant le rôle de support de l'information génétique (avant l'apparition de l'ADN). Dès lors, une étape de traduction est nécessaire afin de synthétiser les protéines encodées par de l'ARN.

La traduction est un processus généralement fiable. Différentes erreurs peuvent cependant se produirent, bien que restant relativement rares. La probabilité qu'un codon soit traduit en un mauvais acide aminé (faux-sens) est de l'ordre de 5×10^{-4} [Par89]. La probabilité que la traduction se termine prématurément est de l'ordre de 10^{-5} [Jor93]. La probabilité de perte du cadre de lecture est estimée à moins de 3×10^{-5} par codon [Kur92]. L'impact des erreurs de traduction d'un codon est généralement limité. En effet, seul certains acides aminés sont indispensables à la structure de la protéine. La plupart peuvent être remplacés par des acides aux propriétés physico-chimiques similaires. Les 2 autres types d'erreurs sont quand à eux fréquemment délétères et provoque généralement la synthèse de protéines non fonctionnelles. Lors d'une perte de la phase de lecture, toute la portion de la protéine située après la protéine est altérée et un codon stop est alors souvent rencontrée après la lecture de quelques dizaines de trinucléotides.

Les erreurs de traduction décrites précédemment sont aléatoires et généralement rares. Un type particulier d'erreur de traduction a pu être mis en évidence. En effet, certains décalages de phase de lecture lors de la traduction se produisent de façon fréquente, entrant en compétition avec le décodage classique. Ils permettent de synthétiser des protéines fonctionnelles. Ce type d'événement est appelé erreurs de traduction programmées ou recodage.

1.5.2 Recodage

Le terme de recodage [GWA92] désigne tout ce qui n'est pas décodage classique de l'ARNm : départ d'un site d'initiation, lecture de codons et traduction en acides aminés (parmi les 20 acides aminés traditionnels) jusqu'à un site de terminaison. Le recodage est

donc une erreur de traduction, se produisant fréquemment et dépendant d'un site d'action spécifique de l'ARN messager. Le recodage est une alternative de décodage et est donc par conséquent en compétition avec la lecture traditionnelle. La protéine synthétisée par la lecture classique n'est pas nécessairement fonctionnelle. Ce qui distingue les erreurs de traduction du recodage est que la protéine synthétisée par recodage est fonctionnelle.

Les premiers exemples de recodage ont été observés par expérimentation. Les recodages ont été catégorisés en fonction du type de modification de la traduction. Ainsi, plusieurs cas de gènes définis par un décalage de phase en +1 ou en -1 ont été identifiés. Pour certains gènes, la lecture se poursuit même après un codon stop (translecture du codon stop), le codon stop pouvant dans certains cas être traduit en acides aminés non traditionnels (sélénocistéine [Cop03] et pyrrolysine [IS04]). Un cas plus rare est le saut de ribosome : à partir d'un site de l'ARN messager, le ribosome "décolle" et cesse donc la lecture d'une séquence de nucléotides avant de poursuivre sa lecture plus loin.

Le recodage a été initialement observé dans les virus ([Far95], [Hun98]). Actuellement, bien que restant rares, des processus de recodage ont pu être observés chez les archaea [CRM05], bactéries ([Bar03], [Nam04]) ou eucaryotes ([Sta01], [Nam04]).

Chaque génome de bactérie a sa propre distribution de trinucléotide [GGG80]. L'usage des codons synonymes (trinucléotides codant pour un même acide aminés) est biaisés : un sous-ensemble réduit de codons est préféré dans les gènes. L'utilisation de codons est corrélée à l'expressivité des gènes [Sha05]. Une explication possible est que l'usage des codons reflète la variation de concentration des ARNt. Les codons majoritaires, encodés par les ARNt les plus abondants, permettraient d'améliorer l'efficacité de la traduction [Bul91]. Néanmoins, l'abondance des ARNt pourrait aussi avoir évolué pour correspondre aux apparitions des codons dans les génomes et serait alors plutôt une conséquence du biais des codons synonymes. Une recherche de signal remarquable a été effectuée par l'analyse des fréquences des codons. La méthode choisie dans un premier temps pour étudier les occurrences d'ensembles de motifs pouvant apparaître dans les gènes est la fonction d'autocorrélation.

Chapitre 2

Fonctions d'autocorrélation appliquées au gènes

2.1 Recherche de motifs à trous dans les gènes

Le concept de motif s'introduit naturellement d'un point de vue biologique. Les gènes (mots) sont formés d'une suite de nucléotides (lettres). La suite de nucléotides dans un gène est la conséquence de plusieurs facteurs biologiques : de son processus de construction (par exemple, la concaténation d'un oligonucléotide), de son processus d'évolution (substitutions, insertions et délétions de nucléotides, distance d'édition, etc.), de ses contraintes spatiales (ADN en double hélice, ARN de transfert en feuille de trèfle refermée en forme de L, etc.) et de sa fonction (gènes codant les protéines, gènes codant les ARN). Ces différents facteurs impliquent que les nucléotides n'ont pas tous la même importance dans ces suites et donc, l'existence de suites non-aléatoires appelées motifs. Le facteur évolutif à l'origine de ces motifs revêt un intérêt tout particulier.

Comprendre ces suites de nucléotides revient donc d'une certaine façon à étudier les motifs qui leurs sont associés. Par exemple, la recherche de gènes dans le génome (début, fin, sens et phase) peut être réalisée en déterminant les motifs associés à ces gènes : codon d'initiation, codons de terminaison, promoteurs, sites d'épissage, motifs de positionnement des histones et de courbure de l'ADN, motifs de structure secondaire de l'ARN messager, dinucléotides CG, codons préférentiels associés au génome, à une fonction ou à une structure secondaire protéique (hélice α, feuillet β), etc. La notion de codons préférentiels résulte de l'utilisation de certains codons pour un acide aminé donné grâce à la dégénérescence du code génétique et de l'utilisation de certains acides aminés grâce à leurs similarités physico-chimiques.

Au concept biologique de motif peut être associée une définition informatique du motif. Un motif est un mot sur un alphabet biologique. L'alphabet génétique est formé de 4 lettres (nucléotides ou bases) : A=Adénine, C=Cytosine, G=Guanine et T=Thymine. Les alphabets génétiques réduits ont également une fonction importante en biologie, en particulier l'alphabet purine/pyrimidine (R=purine=A ou G, Y=pyrimidine=C ou T) car il serait l'alphabet des gènes primitifs ([CG57], [ES78]). L'autre alphabet biologique résultant des processus de transcription et de traduction de l'ARN est l'alphabet protéique formé de 20 lettres. Il existe de nombreux alphabets protéiques réduits qui sont déterminés par les propriétés physico-chimiques associées aux acides aminés : hydrophobicité, volume, poids moléculaire, structure secondaire, etc.

Pour un alphabet donné, un motif est caractérisé par plusieurs paramètres : sa longueur (de l'ordre de la dizaine de lettres), sa structure (motif simple, motif à trous, motif répété, motif palindromique, motif complémentaire, etc.), sa forme géométrique (tige, boucle, etc.), sa localisation, sa fréquence d'occurrence (dans un gène ou une population de gènes) et son histoire évolutive (en fonction du temps). La définition du motif est donc très générale, expliquant ainsi la grande variété des méthodes informatiques développées.

Nous donnons quelques exemples de motifs identifiés dans les gènes : RNY [She81] et GNN [Tri87] (N=A, C, G ou T) dans les gènes, CAAAAT et TTGACA séparé de 17 lettres de TATAAT pour le promoteur de *E. Coli*, AGGAGGT quelques lettres avant le codon d'initiation ATG des gènes des procaryotes [SD74], AGGTRAGT pour le site d'épissage 5', YYTTYYYYYYNCAGG pour le site d'épissage 3', CTRAY pour le site de branchement du "lasso" [SS86].

La fonction d'autocorrélation est un outil puissant pour la recherche de motifs dans les gènes, en particulier les motifs à trous. Elle permet en particulier d'étudier des périodicités et des pics d'occurrences.

2.2 Fonction d'autocorrélation

2.2.1 Fonction d'autocorrélation moyenne

Soient $\mathcal{F}$ un langage sur un alphabet fini et w et w' 2 mots de $\mathcal{F}$. La fonction $i \rightarrow A_{w,w'}(i, \mathcal{F})$ donnant la probabilité que w' apparaisse i lettres quelconques après w dans $\mathcal{F}$, est appelée fonction d'autocorrélation. Dans le cas d'un langage associé à des gènes, plusieurs méthodes permettent de calculer le résultat de la fonction d'autocorrélation.

Les méthodes classiques utilisent le spectre de puissance qui est la transformée de Fourier de la fonction d'autocorrélation. Cette approche présente 2 inconvénients dans l'étude des gènes. D'une part, le spectre de puissance est en bijection avec une fonction d'autocorrélation qui est décroissante quand i croît. D'autre part, le spectre de puissance permet d'identifier une périodicité, par exemple modulo 3, mais pas son type 0, 1 ou 2 modulo 3. Par exemple, la fonction d'autocorrélation peut traduire une périodicité 0 modulo 3 par des maximums pour $i = 0, 3, 6$, etc., une périodicité 1 modulo 3, par des maximums pour $i = 1, 4, 7$, etc., et une périodicité 2 modulo 3, par des maximums pour $i = 2, 5, 7$, etc. , alors que le spectre de puissance montre un même pic en 0.33 pour ces 3 types de périodicités modulo 3. La fonction d'autocorrélation est donc plus simple que le spectre de puissance pour l'analyse de certaines informations contenues dans les gènes, comme le maximum global, les maximums locaux et les décalages de périodicité.

Nous donnons une méthode de calcul de la fonction d'autocorrélation qui évite la décroissance des probabilités par correction de l'effet de bord induit par la fin du gène (perte de la bijection avec le spectre de puissance) pour des mots w et w' de longueurs quelconques sur l'alphabet des gènes à 4 lettres.

Une population $\mathcal{F}$ de gènes est constituée de n($\mathcal{F}$) gènes sur l'alphabet génétique $\mathcal{A}$ = {A, C, G, T}. Soit s un gène de $\mathcal{F}$ de longueur $l(s)$. Soient 2 motifs w et w' de longueurs respectives $|w|$ et $|w'|$ sur l'alphabet $\mathcal{A}$. Soit m_i, appelé i-motif, 2 motifs w et w' séparés par $i (i \in [0, i_{max}])$ bases quelconques N et noté $m_i = w(N)_i w'$. Pour chaque gène s de $\mathcal{F}$, le compteur $c_i(s)$ compte les occurrences de m_i dans s. Pour compter les occurrences de m_i dans les mêmes conditions pour tout $i \in [0, i_{max}]$, uniquement les $L(s) = l(s) - (i_{max} + |w| + |w'|) + 1$ premiers nucléotides de s sont considérés. La probabilité d'occurrence $p_i(s)$ de m_i dans s est égale au ratio du compteur par le nombre de nucléotides étudiés : $p_i(s) = c_i(s)/L(s)$. La probabilité d'occurrence $A\{w, w'\}(i, \mathcal{F})$ de m_i dans $\mathcal{F}$ est donc égale à

$$A_{w,w'}(i,\mathcal{F}) = \frac{1}{n(\mathcal{F})}\sum\nolimits_{s\in\mathcal{F}} p_i(s)$$

2.2.2 Fonction d'autocorrélation en phase

Une étude statistique complète des gènes a conduit à généraliser la définition de la fonction d'autocorrélation $A_{w,w'}(i,\mathcal{F})$ aux 64 trinucléotides $w, w' \in= \{AAA, ..., TTT\}$ en considérant pour w les 3 phases $p \in 0, 1, 2$ des gènes codants, $p = 0$: phase de référence établie par le codon d'initiation ATG et $p = 1$ (resp. $p = 2$) phase de référence décalée de 1 (resp. 2) base dans la direction 5'-3' (vers la droite). Un trinucléotide w en phase p est noté w^p. La fonction $i \rightarrow A_{w^p,w'}(i,F)$ donnant la probabilité que w' en phase quelconque apparaisse i bases quelconques après w^p en phase p dans la population de gènes $\mathcal{F}$ est appelée fonction d'autocorrélation en phase $w^p(N)_i w'$. Ainsi, $64^2 \times 3 = 12288$ fonctions d'autocorrélation $A_{w^p,w'}(i,\mathcal{F})$ sont associées aux i-motifs $w^p(N)_i w' \in \{AAA^p(N)_i AAA, \ldots, TTT^p(N)_i TTT\}$ avec w^p en phase p et w' en phase quelconque.

La méthode de calcul de la fonction d'autocorrélation en phase est identique à la méthode de calcul de la fonction d'autocorrélation moyenne des 3 phases donnée dans la section 2.2.1., sauf que le nombre $L(s)$ de nucléotides étudiés dans la séquence s est divisé par 3 (w^p est dans une des 3 phases).

La fonction $i \rightarrow A_{w^p,w'}(i,F)$ donnant la probabilité d'occurrence que w' apparaisse i bases quelconques après w^p dans la population de gènes $\mathcal{F}$, est dite fonction d'autocorrélation $w^p(N)_i w'$ (associée au i-motif $w^p(N)_i w'$). Cette fonction d'autocorrélation $w^p(N)_i w'$ est représentée par une courbe avec en abscisse, le nombre i de bases N entre w et w', i variant de 0 à 50 (imax=50 en général) et avec en ordonnée, la probabilité $A_{w^p,w'}(i,\mathcal{F})$ d'occurrence de $w^p(N)_i w'$ dans la population de gènes $\mathcal{F}$. La fonction d'autocorrélation ainsi définie possède les propriétés suivantes :

(i) sans biais : correction de l'effet de bord induit par la fin du gène

(ii) simple : basée sur les probabilités

(iii) avec une représentation graphique facilement interprétable : identification de divers types de périodicités, etc ;

(iv) générale : un motif est un cas particulier d'un i-motif avec i=0 ; un gène est un cas particulier d'une population de gènes et la définition donnée pour un alphabet à 4 lettres peut être prolongée à un alphabet quelconque

(v) statistiquement stable : calcul des probabilités au niveau des populations de gènes (conséquence de la loi des grands nombres).

2.2.3 Algorithmes de calcul de la fonction d'autocorrélation

Nous mentionnons les principaux algorithmes développés pour calculer la fonction d'autocorrélation dans différentes situations de complexité comme l'identification de propriétés statistiques non-aléatoires dans les populations de gènes, les modèles d'évolution avec un processus de construction et de mutation des gènes. Deux principaux algorithmes calculent la fonction d'autocorrélation dans les populations de gènes :

(i) un algorithme de comptage d'occurrence des i-motifs par parcours de séquences, simple mais seulement utilisable pour quelques fonctions d'autocorrélation.

(ii) un algorithme de comptage d'occurrence des i-motifs par adresse pour calculer des milliers de fonctions d'autocorrélation ayant des motifs w et w' de longueurs élevées (il existe $4^{10} > 10^6$ motifs de longueur 10 sur A,C,G,T) et/ou des i-motifs séparés d'un nombre i élevé de bases ($i \geq 100$). La base de données de gènes est transformée en une base de données d'adresses de motifs pour éviter de tester tous les motifs dans toutes les positions des séquences.

2.3 Identification de périodicités décalées

Les 12288 fonctions d'autocorrélation en phase $w^p(N)_i w'$ ont été appliquées aux gènes de procaryotes F =PRO (13686 séquences) et d'eucaryotes F =EUK (26757 séquences). La comparaison des fonctions d'autocorrélation obtenues a permis de distinguer 3 classes de périodicités. En appliquant une simple règle (cf. ci-dessous), elles permettent d'associer chaque trinucléotide à une phase d'occurrence préférentielle.

Les 3 types de périodicité identifiés sont :

(i) Type 0 : périodicité 0 modulo 3 avec des valeurs maximales en i=0, 3, 6, etc.

(ii) Type 1 : périodicité 1 modulo 3 avec des valeurs maximales en i=1, 4, 7, etc.

(iii) Type 2 : périodicité 2 modulo 3 avec des valeurs maximales en i=2, 5, 8, etc.

La méthode permettant de classer une fonction d'autocorrélation $w^p(N)_i w'$ ($w, w' \in F$, $p \in \{0,1,2\}$ et $i \in [0,99]$) dans un type de périodicité consiste à déterminer pour chaque type de périodicité modulo 3, le nombre de points qui sont supérieurs à leurs 2 points adjacents. Pour une courbe de 100 points, le nombre maximum de points supérieurs à ses 2 points adjacents est de 33. Si le nombre de points pour un type p de périodicité est supérieur à 22 points (seuil de significativité obtenu par un test binomial) alors la fonction d'autocorrélation est de type p. De plus, avec un tel seuil, une fonction d'autocorrélation ne peut appartenir qu'à un seul type. Quelques fonctions d'autocorrélation ne sont pas classables, en particulier les fonctions d'autocorrélation avec un mot w associé à un codon de terminaison ont des probabilités nulles pour tout i.

Les 3 types de périodicité modulo 3 existent quelque soit la phase p du trinucléotide w ($w \in F$, $p \in \{0,1,2\}$). En effet, la périodicité 0 modulo 3 peut être observée en phase 0, en phase 1 et en phase 2. De façon similaire, la périodicité 1 (resp. 2) modulo 3 peut être observée en phase 0, en phase 1 et en phase 2. Lorsqu'une fonction d'autocorrélation $w^p(N)_i w'$ a une périodicité j modulo 3, le trinucléotide w' est en phase $q(w,p) = (p+j) mod 3$ après w en phase p. La fonction d'autocorrélation en phase permet donc, en comparant les périodicités, d'associer chaque trinucléotide à une phase particulière.

La fonction d'autocorrélation en phase a donc permis d'identifier dans les gènes de procaryotes et d'eucaryotes des ensembles de codons associés préférentiellement à une phase. La recherche de trinucléotides associés préférentiellement à une phase peut être réalisée plus simplement par la méthode décrite dans la section suivante.

2.4 Méthode FTF

La méthode Frame Trinucleotide Frequency (FTF) permet d'identifier statistiquement une phase d'occurrence préférentielle pour un motif (codon). Elle donne donc le même type de résultat que la fonction d'autocorrélation en phase mais est plus simple d'utilisation. Dans l'alphabet génétique, il existe 64 trinucléotides $w \in \{\text{AAA}, \ldots, \text{TTT}\}$. Dans les gènes, un trinucléotide w peut être trouvé dans 3 phases $p \in \{0,1,2\}$. Il existe donc $64 \times 3 = 192$ trinucléotides w^p. Soit $\sigma(w^p)$ la fréquence d'occurrence du trinucléotide w dans la phase p dans un gène. La phase d'occurrence préférentielle de w est égale à k si

$$\sigma(w^k) = MAX_{p=0}^{2}\sigma(w^p)$$

Le maximum n'est pas nécessairement unique. Lorsque l'écart entre les plus fortes valeurs des fréquences d'occurrence d'un trinucléotide dans plusieurs phases sont égales sont inférieur à une certaine valeur ϵ, le choix de la phase préférentielle du trinucléotide est ambigu. Il s'agit des exceptions de la méthode FTF.

De cette manière, chaque trinucléotide w est associé a une phase d'occurrence préférentielle, la phase avec la plus haute fréquence d'occurrence.

2.5 Motifs associés préférentiellement à une phase

De façon inattendue, on trouve dans les 2 populations de gènes étudiées que les 64 trinucléotides w ont une phase d'occurrence préférentielle constante et peuvent être classés en 3 sous-ensembles de trinucléotides selon la phase. Les 22 trinucléotides en phase 0 forment le sous-ensemble $\mathcal{T}_0$ et les 21 trinucléotides dans chacune des phases 1 et 2, les sous-ensembles $\mathcal{T}_1$ et $\mathcal{T}_2$ respectivement (voir table 2.1).

$\mathcal{T}_0$	*AAA* AAC AAT ACC ATC ATT CAG CTC CTG GAA GAC GAG GAT GCC GGC GGT GTA GTC GTT TAC TTC *TTT*
$\mathcal{T}_1$	AAG ACA ACG ACT AGC AGG ATA ATG CCA *CCC* CCG GCG GTG TAG TCA TCC TCG TCT TGC TTA TTG
$\mathcal{T}_2$	AGA AGT CAA CAC CAT CCT CGA CGC CGG CGT CTA CTT GCA GCT GGA *GGG* TAA TAT TGA TGG TGT

TAB. 2.1 – Ensembles des trinucléotides $\mathcal{T}_0$, $\mathcal{T}_1$ et $\mathcal{T}_2$ associés respectivement aux phases 0, 1 et 2.

En ôtant les 4 trinucléotides avec 3 lettres identiques des 3 sous-ensembles $\mathcal{T}_0$, $\mathcal{T}_1$ et $\mathcal{T}_2$ les 3 sous-ensemble $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$ sont définis : $\mathcal{X}_0 = \mathcal{T}_0 \backslash \{\text{AAA,TTT}\}$, $\mathcal{X}_1 = \mathcal{T}_1 \backslash \{\text{CCC}\}$ et $\mathcal{X}_2 = \mathcal{T}_2 \backslash \{\text{GGG}\}$ associés à la phase 0, 1 et 2 respectivement.

Des propriétés intéressantes sont trouvées pour les sous-ensembles $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$.

Permutation circulaire

La permutation circulaire P d'un trinucléotide $w = a_1a_2a_3$ est le trinucléotide permuté $P(w) = a_2a_3a_1$. Les ensembles $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$ sont équivalents par permutation circulaire. On a $\mathcal{X}_1 = \{\ P(a_1a_2a_3) = a_2a_3a_1 | a_1a_2a_3 \in \mathcal{X}_0\ \}$. De la même manière, $\mathcal{X}_2$ peut être obtenu par permutation circulaire de $\mathcal{X}_1$ et $\mathcal{X}_0$ par permutation circulaire de $\mathcal{X}_2$ (voir table 2.2). Chacun des ensembles peut donc être déduit des deux autres. L'ensemble $\mathcal{X}_0$ contenant les codons associés à la phase de lecture est généralement pris comme représentant des 3 ensembles. Il peut être noté $\mathcal{X}$ ($\mathcal{X}$=$\mathcal{X}_0$).

$\mathcal{X}_0$	AAC	AAT	ACC	ATC	CAG	CTC	GAA	GAC	GCC	GTA	GTT	ATT	GGT	GAT	CTG	GAG	TTC	GTC	GGC	TAC
$\mathcal{X}_1$	ACA	ATA	CCA	TCA	AGC	TCC	AAG	ACG	CCG	TAG	TTG	TTA	GTG	ATG	TGC	AGG	TCT	TCG	GCG	ACT
$\mathcal{X}_2$	CAA	TAA	CAC	CAT	GCA	CCT	AGA	CGA	CGC	AGT	TGT	TAT	TGG	TGA	GCT	GGA	CTT	CGT	CGG	CTA

TAB. 2.2 – Les ensembles $\mathcal{X}_0$, $\mathcal{X}_1$, $\mathcal{X}_2$ sont équivalents par permutation circulaire.

Complémentarité

Rappel de la règle de complémentarité de l'ADN :
i)La double hélice d'ADN est formée de 2 séquences (chaînes) de nucléotides $s1$ et $s2$ appariées selon la règle de complémentarité $\mathcal{C}$: le nucléotide A (resp. C, G, T) dans $s1$ apparié avec le nucléotide complémentaire $\mathcal{C}$(A)=T (resp. $\mathcal{C}$(C)=G, $\mathcal{C}$(G)=C, $\mathcal{C}$(T)=A) dans $s2$.
ii)Les 2 séquences de nucléotides $s1$ et $s2$ sont appariées dans la double hélice d'ADN ont des directions opposées (anti-parallèle) : le trinucléotide $w = a1a2a3$ dans $s1$ est apparié avec le trinucléotide complémentaire $\mathcal{C}$(w)= $\mathcal{C}$(a3)$\mathcal{C}$(a2)$\mathcal{C}$(a1) dans $s2$.

L'observation des ensembles de trinucléotides préférentiels permet de remarquer que $\mathcal{C}(\mathcal{X}_0)= \mathcal{X}_0$, $\mathcal{C}(\mathcal{X}_1)= \mathcal{X}_2$ et $\mathcal{C}(\mathcal{X}_2)= \mathcal{X}_1$. Par conséquent, $\mathcal{X}_0$ est auto-complémentaire et $\mathcal{X}_1$ et $\mathcal{X}_2$ sont complémentaires entre eux (voir table 4.3).

$\mathcal{X}_0$	AAC	AAT	ACC	ATC	CAG	CTC	GAA	GAC	GCC	GTA										
$\mathcal{X}_0$	GTT	ATT	GGT	GAT	CTG	GAG	TTC	GTC	GGC	TAC										
$\mathcal{X}_1$	AAG	ACA	ACG	ACT	AGC	AGG	ATA	ATG	CCA	CCG	GCG	GTG	TAG	TCA	TCC	TCG	TCT	TGC	TTA	TTG
$\mathcal{X}_2$	CTT	TGT	CGT	AGT	GCT	CCT	TAT	CAT	TGG	CGG	CGC	CAC	CTA	TGA	GGA	CGA	AGA	GCA	TAA	CAA

TAB. 2.3 – L'ensemble $\mathcal{X}_0$ est auto-complémentaire, $\mathcal{X}_1$ est complémentaire à $\mathcal{X}_2$.

De plus, l'étude des ensembles $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$ montre que ces ensembles possèdent des propriétés en tant que code : ils forment des codes circulaires. Dans le chapitre suivant, des notions générales de théorie des codes seront données. Plusieurs classes de codes possédant différentes capacités de décodages seront présentées. Elles permettront de caractériser les codes pouvant être associés aux ensembles $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$. En particulier, les différentes propriétés des codes circulaires seront décrites.

Chapitre 3

Elements de théorie des codes

Dans cette partie, nous décrivons quelques notions de bases sur les codes. La représentation algébrique des codes est issue de travaux de M. P. Schützenberger [Sch56]. L'étude des codes fait appel à des notions issues de diverses disciplines de l'informatique comme la combinatoire des mots, la théorie des monoïdes et la théorie des graphes, plus précisément des automates. En nous appuyant sur divers ouvrages et articles, et en particulier sur l'ouvrage de référence, *Theory of codes* de Berstel J. et Perrin D. [BP85], nous proposons une synthèse des définitions, propriétés et théorèmes en rapport avec les codes circulaires, classe de code que nous avons identifiée dans les gènes.

3.1 Notions de base

3.1.1 Mots

Définitions

Soit $\mathcal{A}$ un ensemble non vide appelé alphabet. Les éléments de $\mathcal{A}$ sont des lettres. Un *mot* w sur l'alphabet $\mathcal{A}$ est une suite finie de lettres

$$w = (l_0, l_1, l_2, \ldots, l_n), l_i \in \mathcal{A}$$

L'ensemble de tous les mots sur $\mathcal{A}$ est noté $\mathcal{A}^* \cdot \mathcal{A}$ est muni d'une loi de composition interne appelée concaténation, généralement représenté par le symbole $\cdot$ et définie comme suit

$$(a_0, a_1, a_2, \ldots, a_n)(b_0, b_1, b_2, \ldots, b_m) = (a_0, a_1, a_2, \ldots, a_n, b_0, b_1, b_2, \ldots, b_m)$$

Cette opération est associative. Par conséquent, afin d'alléger les notations, l'écriture suivante est utilisée

$$w = l_0 l_1 l_2 \ldots l_n$$

Pour tout mot w de $\mathcal{A}$, la longueur de w est le nombre d'occurrences de lettres de $\mathcal{A}$ dans w, elle est notée $|w|$. L'unique élément neutre de $\mathcal{A}^*$ muni de la concaténation est le mot de longueur 0, il est appelé mot vide et est noté ϵ ou 1.

L'ensemble des mots non vides de $\mathcal{A}$ est noté $\mathcal{A}^+$. On a donc $\mathcal{A}^+ = \mathcal{A}^* \setminus \epsilon$.

Pour une lettre l de $\mathcal{A}$ et un mot w de $\mathcal{A}^*$, nous notons $|w|_l$ le nombre d'occurrences de la lettre l dans le mot w. Nous notons $alph(w)$ l'ensemble des lettres apparaissant dans le mot w : $alph(w) = \{l \in \mathcal{A} \mid |w|_l > 0\}$

Pour tous $X \subset \mathcal{A}^*$, la notation précédente est généralisée au langage

$$alph(X) = \bigcup_{x \in X} alph(x)$$

Etant donné un sous-ensemble non vide X de $\mathcal{A}^*$, la cardinalité de X est noté $\mathrm{Card}(X)$.

3.1.2 Facteurs, préfixes et suffixes

Définitions

Soit $x \in \mathcal{A}^+$. Un mot $w \in \mathcal{A}^*$ est un *facteur* de x s'il existe $u, v \in \mathcal{A}^*$ tel que $x = uwv$. Il existe donc une occurrence de w dans x, sa position est définie comme étant égale à $|u|+1$. Le mot w est un *facteur propre* de x si $w \neq x$. Un facteur est non trivial s'il est différent du mot vide.
Le mot w est un *préfixe* (resp. *suffixe*) de x si $u = \epsilon$ (resp. $v = \epsilon$).

Soient $w \in \mathcal{A}^*$ et X un ensemble. Le mot $u \in X$ est un X-facteur de w s'il existe $u_1, u_2 \in \mathcal{A}^*$ tels que $w = u_1 \cdot u \cdot u_2$. De plus, u est un X-préfixe (resp. X-suffixe) de w si $u_1 = \epsilon$ (resp. $u_2 = \epsilon$).

L'ensemble des facteurs (resp. préfixes, suffixes) de w est noté $F(w)$ (resp. $P(w)$, $S(w)$).

Etant donnée une partie $\mathcal{A} \in \mathcal{A}^*$, on note

$$F(X) = \bigcup_{w \in X} F(w)$$

$$P(X) = \bigcup_{w \in X} P(w)$$

$$S(X) = \bigcup_{w \in X} S(w)$$

En outre, nous notons $\overline{P(X)}$ (resp. $\overline{S(X)}$) l'ensemble $P(X) \setminus \{\epsilon\}$ (resp. $S(X) \setminus \{\epsilon\}$).

Un mot $w \in \mathcal{A}^*$ est sans bord si aucun de ses préfixes propres non triviaux n'est suffixe de w. En d'autres termes, w ne se chevauche pas avec lui-même (fig. 3.1).

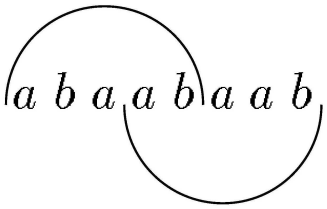

FIG. 3.1 – Le mot abaabaab n'est pas sans bord

La notion de *permutation circulaire* d'un mot est utilisé à plusieurs reprises dans ce travail. Nous en donnons une définition.

Définition : Permutation circulaire

Soit le mot $w = w_0 w_1 ... w_n$. Le voisin gauche (resp. droit) d'une lettre w_i avec $i > 0$ (resp. $i < n$) est la lettre w_{i-1} (resp. w_{i+1}), le voisin gauche de w_0 est w_n et le voisin droit de w_n est w_0.
Un mot w' est un permuté circulaire de w si w' contient les mêmes lettres que w et ces lettres ont les mêmes voisins gauches et droits dans les deux mots.

3.2 Monoïdes

3.2.1 Définitions

Un *monoïde* est un ensemble muni d'une loi de composition interne associative et qui possède un élément neutre. La loi de composition est représentée par le symbole de multiplication (l'élément neutre est unique et sera noté 1).

Soit deux parties X et Y d'un monoïde M , $X \cdot Y$ désigne le produit suivant

$$X \cdot Y = \{x \cdot y \in M \mid x \in X, y \in Y\}$$

De plus, les ensembles $X.Y^{-1}$ et X^{-1}.Y sont définis de la façon suivante

$$X.Y^{-1} = \{z \in M \mid \exists (x, y) \in X \times Y, x = z.y\}$$

$$X^{-1}.Y = \{z \in M \mid \exists (x, y) \in X \times Y, x.z = y\}$$

L'ensemble des préfixes (resp. suffixes) propres des mots de X est noté $X\mathcal{A}^-$ (resp. $\mathcal{A}^-X$) tel que $w \in X\mathcal{A}^- \Rightarrow \exists x \in X, u \in \mathcal{A}^+, x = wu$.

Etant donné un monoïde M, un sous-ensemble N de M est un sous-monoïde de M s'il contient l'élément neutre et est stable pour la loi de composition interne :

$$N \cdot N \in N$$

$$1_M \in N$$

Un sous-ensemble N de M peut satisfaire la première condition sans satisfaire la seconde ($1_m \neq 1_n$). N est un monoïde sans être un sous-monoïde de M.

3.2.2 Monoïde syntaxique

Des liens existent entre les monoïdes et la combinatoire des mots, $\mathcal{A}^*$ muni de l'opération associative de concaténation est un monoïde.

Définition

Soit $X \subset \mathcal{A}^*$. Pour tout mot $w \in \mathcal{A}^*$, nous notons

$$\Gamma(w) = \{(u, v) \in \mathcal{A}^* \times \mathcal{A}^* \mid uwv \in X\}$$

La congruence syntaxique de X est la relation d'équivalence σ_X définie par :

$$w \equiv \sigma_X w' \Rightarrow \Gamma(w) = \Gamma(w')$$

Le quotient de $\mathcal{A}^*$ par σ_X est par définition le monoïde syntaxique que nous notons $M(X)$. Un ensemble X est dit *reconnaissable* si $M(X)$ est fini.

3.3 Codes

3.3.1 Définition

Une partie non vide $X \in \mathcal{A}^+$ est un *code* sur $\mathcal{A}$ si pour tout $n, m \geq 1$ et tous mots $x_1, \ldots, x_n \in X$ et $x'_1, \ldots, x'_m \in X$, la condition $x_1 x_2 \ldots x_n = x'_1 x'_2 \ldots x'_m$ implique $n = m$ et $x_i = x'_i$ pour $i \in [1, n]$. Tout mot de $\mathcal{A}^*$ a donc au plus une factorisation en mots de X.

La proposition suivante est une conséquence directe de cette définition :

Proposition

Tout sous-ensemble d'un code est un code.

Considérons deux exemples

Exemple 1

L'ensemble $X = \{ab, ba\}$ sur l'alphabet $\mathcal{A} = \{a, b\}$ est un code. En effet, soit un mot $w \in X^+$. Prenons $w = w'x$ avec $w' \in X^*$ et $x \in X$. Si w se termine par a, on a $x = ba$ sinon $x = ab$. On factorise w' de la même manière jusqu'à ce que $w' = ab$ ou $w' = ba$. La factorisation de tout mot w en mots de X est donc unique.

Exemple 2

L'ensemble $X = \{a, ab, ba\}$ sur l'alphabet $\mathcal{A} = \{a, b\}$ n'est pas un code. En effet, il existe un mot $w = aba$ qui dispose de deux factorisations distinctes en mots du code : $w = (ab)a = a(ba)$

On remarque que l'alphabet $\mathcal{A}$ est un code.

Un code composé d'un ensemble fini d'éléments est appelé *code fini*.

Un *code uniforme* est un code dont tous les éléments sont de même longueur.

3.3.2 Classes de code classiques

Un code $X \subset \mathcal{A}^+$ est préfixe (resp. suffixe) si $X \cap X\mathcal{A}^+ = \emptyset$ (resp. $X \cap \mathcal{A}^+ X = \emptyset$).
Un code préfixe et suffixe est *bifixe*.

Exemple 3

Soit les ensembles $X = \{ba, a\}$, $Y = \{ab, b\}$ et $Z = \{ba, ab\}$. X est un code préfixe (mais pas suffixe), Y est un code suffixe (mais pas préfixe) et Z est un code préfixe et suffixe donc bifixe.

Les codes à délai de déchiffrage fini généralisent la notion de code préfixe. Soit X un ensemble, la concaténation de n mots de X est noté X^n.

Définition [GM59]

Un ensemble $X \subset \mathcal{A}^+$ est un *code à délai de déchiffrage fini* s'il existe un entier $d \geq 0$ tel que pour tout $x, x' \in X, y \in X^d, u \in \mathcal{A}^*$ on a :

$$xyu \in x'X^* \Rightarrow x = x' \qquad (1.1)$$

Le plus petit entier d vérifiant (1.1) est le délai de déchiffrage de X. Les codes à délai de déchiffrage fini imposent de lire d mots avant d'être sur du décodage qui suit. Un code préfixe est donc à délai de déchiffrage 0.

Exemple 4

L'ensemble $X = \{a, ab\}$ n'est pas un code préfixe mais est un code à délai de déchiffrage fini avec un délai de 1.

Exemple 5

L'ensemble $X = \{aa, ba, a\}$ a un délai de déchiffrage infini. Pour tout $d \geq 0$, le mot $b(aa) \in X^{1+d}$ est un facteur de $ba(aa)$.

Il existe d'autre notion de délai de déchiffrage [Bru91].

3.3.3 Monoïde libre

Soit M un sous-monoïde de $\mathcal{A}^*$. L'ensemble minimal de générateurs X de M est l'ensemble (unique) :

$$X = (M \setminus \{\epsilon\}) \setminus (M \setminus \{\epsilon\})^2$$

Un sous-monoïde M est dit libre si son ensemble minimal de générateurs est un code. Réciproquement, si X est un code, alors l'ensemble minimal de générateurs de X^* est X. Dans ce cas, X^* est un sous-monoïde libre. Le code X est appelé la base de X^*.

Soit M un monoïde. Un sous-monoïde N de M est stable dans M si pour tout $m, m', x \in M$ on a :

$$m, m', mx, xm' \in N \Rightarrow x \in N$$

La condition de stabilité peut aussi être écrite sous la forme :

$$N^{-1}N \cap NN^{-1} = N$$

Un sous-monoïde $M \subset \mathcal{A}^*$ est libre si et seulement si il est stable [Sch56].

3.3.4 Algorithme de Sardinas et Patterson

L'algorithme de Sardinas et Patterson est un procédé classique qui permet de décider si un ensemble reconnaissable est un code :

Soit $X \in \mathcal{A}^+$, définissons la suite $(U_n)_{n\geq 1}$ par :

$$U_1 = X^{-1}X \setminus \{\epsilon\}$$

$$U_{n+1} = X^{-1}U_n \cap U_n^{-1}X \text{ pour } n \geq 1$$

Proposition [SP53]

L'ensemble $X \in \mathcal{A}^+$ est un code si et seulement si aucun ensemble U_n ne contient le mot vide. De plus, si X est reconnaissable, alors l'ensemble des U_n est fini.

3.4 Factorisations et interprétations

Soit $w \in \mathcal{A}^*$. Une *factorisation* de w est une suite $(u_1, u_2, \ldots, u_n)$, $n \geq 0$ et $u_i \in \mathcal{A}^*$ de mots de $\mathcal{A}^*$ tels que $w = u_1 u_2 \ldots u_n$.

Etant donné un sous-ensemble quelconque X de $\mathcal{A}^*$, on appelle *X-factorisation* de $w \in X^*$ toute factorisation de w dont les éléments sont dans X (fig. 3.2).

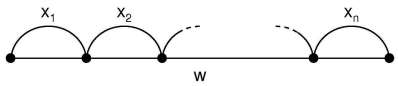

FIG. 3.2 – X-factorisation de w

La notion de *X-interprétation* généralise la notion de X-factorisation : une X-interprétation de w est une suite $(s, d_1, d_2, \ldots, d_n, p)$ telle que $w = sd_1d_2 \ldots d_np$ avec $n \geq 0, s \in \mathcal{A}^-X, p \in X\mathcal{A}^-$ $d_i \in X$ pour $i \in [1, n]$.

Deux X-interprétations $(s, d_1, \ldots, d_n, p)$ et $(s', d'_1, \ldots, d'_m, p')$ du mot w sont *adjacentes* s'il existe $i \in [0, n]$ et $j \in [0, m]$ tels que $sd_1 \ldots d_i = s'd'_1 \ldots d'_j$.
Deux X-interprétations qui ne sont pas adjacentes sont dites *disjointes*.
Une X-interprétation $(\epsilon, d_1, \ldots, d_n, \epsilon)$ est dite *triviale*.
Une X-interprétation $(s, d_1, \ldots d_n, p)$ telle que $s, p \in X^*$ est dite *quasi-triviale*.

Soient $w \in \mathcal{A}^*$ et $u, w', v \in \mathcal{A}^*$ tel que $w = uw'v$ et soit $(d_0, d_1, \ldots, d_n, d_{n+1})$ une X-interprétation de w. La X-interprétation $I = (d_0, d_1, \ldots, d_n, d_{n+1})$ induit une X-interprétation pour l'occurrence (u, w', v) de w' s'il existe $s \in \mathcal{A}^-X, p \in X\mathcal{A}^-$ et $i, j \in \mathbb{N}, 0 < i \leq j \leq n+1$ tels que $s \in S(d_{i-1}), p \in P(d_j)$ et $w' = sd_id_{i+1} \ldots d_{j-1}p$ (voir fig 3.3).

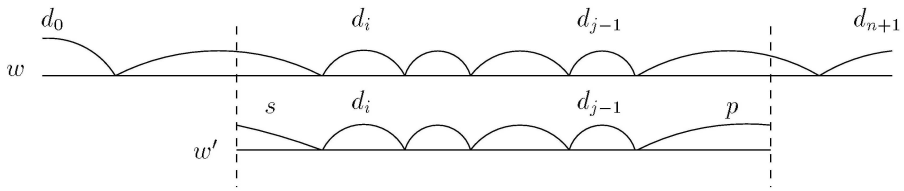

FIG. 3.3 – Une interprétation induite

Dans le cas où X est un code, la donnée du triplet (s, d, p) détermine de façon unique une X-interprétation. La X-interprétation (ϵ, w, ϵ) est alors la X-interprétation triviale de

w (celle-ci est unique). Toute X-interprétation (s, d, p) de w est dite propre si elle n'est pas adjacente à la X-interprétation triviale.

Nous notons $\delta_X(w)$ le nombre maximal d'interprétations deux à deux disjointes du mot w. Le degré du code X, noté $d(X)$, est l'entier

$$d(X) = min\,\{\delta_X(w) \mid w \in X^* \setminus F(X)\}$$

3.5 Automate

3.5.1 Automate simple

Définition

Un automate sur l'alphabet $\mathcal{A}$ est composé de l'ensemble E (ensemble des états), du sous-ensemble $I \subset E$ (ensemble des états initiaux), du sous-ensemble $F \subset E$ (ensemble des états finaux), et du sous ensemble T (ensemble des transitions) défini comme suit $T \subset E \times \mathcal{A} \times E$. Un automate est représenté par le quadruplet (E, I, F, T).

Un automate est fini quand l'ensemble E est fini.

Un chemin dans l'automate Λ est la séquence $p = (f_1, f_2, \ldots, f_n)$ composée de transitions consécutives c'est à dire tel que :

$$f_i = (q_i, a_i, q_{i+1}), i \in [1, n]$$

L'entier n est la longueur de p. L'étiquette du parcours du chemin est $|p| = a_1 a_2 \ldots a_n$. Par convention, si $|p| = w$, le chemin p peut être noté $q_1 \xrightarrow{w} q_{n+1}$.
Un chemin dont l'état initial est identique à l'état final sera appelé un cycle.

Si q_1 est un état initial et q_{n+1} est un état final alors le mot w est reconnu (ou accepté) par l'automate Λ. Le langage reconnu par l'automate Λ est l'ensemble des mots reconnus par Λ, donc :

$$L(\Lambda) = \left\{ w \in \mathcal{A}^* \mid \exists p : i \xrightarrow{w} f, i \in I, f \in F, w = |p| \right\}$$

Les automates peuvent être représentés graphiquement sous la forme d'un multigraphe dont les arêtes et les sommets sont étiquetés.

Un automate est *déterministe* s'il possède un seul état initial et si :

$$(p, a, q), (p, a, q') \in T \Rightarrow q = q'$$

Donc pour tout état $p \in E$ et toute étiquette $a \in \mathcal{A}$, il existe au plus un unique état q tel que $(p, a, q) \in T$.

Un automate est *non ambigu* si il existe au plus un chemin permettant d'aller de p à q étiqueté par w. Les automates non ambigus sont une généralisation des automates non déterministes de la même façon que les codes sont une généralisation des codes préfixes.

3.5.2 Ensembles rationnels

Définition

La famille des ensembles rationnels, notée R, est la plus petite famille des parties de $\mathcal{A}^*$ qui vérifie les trois propriétés suivantes :

- Tout sous-ensemble fini de $\mathcal{A}^*$ appartient à R.
- Si $X, Y \in \mathrm{R}$ alors $X \cup Y \in \mathrm{R}$ et $X \cdot Y \in \mathrm{R}$.
- Si $X \in \mathrm{R}$ alors $X^* \in \mathrm{R}$.

Dans le cas où l'alphabet est fini, le théorème de Kleene établit l'équivalence entre les deux notions de reconnaissabilité et de rationalité :

Théorème (Kleene)

Soit $\mathcal{A}$ un alphabet fini. Un sous-ensemble de $\mathcal{A}^*$ est reconnaissable si et seulement si il est rationnel

3.5.3 Automate à pétales

Les automates à pétales sont des automates pouvant reconnaître des sous-monoïdes de $\mathcal{A}^*$.

Soit X un sous-ensemble de $\mathcal{A}^*$ et l'automate $\mathcal{F}$(X) = (E,I,F,T) avec des états $E = \{(u,v) \in \mathcal{A}^* \times \mathcal{A}^* \mid uv \in X\}$, $I = 1 \times X$, $F = X \times 1$ et des transitions de la forme $(u,v) \xrightarrow{a} (u',v')$ tel que $ua = u'$ et $v = av'$.
Ainsi, l'ensemble des transitions de $\mathcal{F}(X)$ correspond à l'union des transitions associées à chaque $w = l_1 l_2 \dots l_n$ de X. $\mathcal{F}$(X) reconnaît donc X.

L'automate à pétales de X est par définition l'automate $\mathcal{F}^*(X)$ obtenus à partir de $\mathcal{F}(X)$ auquel on ajoute un état supplémentaire, ω, et les transitions suivantes :

- $\omega \xrightarrow{a} (a,v)$ pour $av \in X$,
- $(u,a) \xrightarrow{a} \omega$ pour $ua \in X$,
- $\omega \xrightarrow{a} \omega$ pour $a \in X$.

Les états $1 \times X$ et $X \times 1$ ne sont plus accessibles et disparaissent. Ils sont remplacés par un état unique ω (aussi noté (1,1)). Le graphe $\mathcal{F}$ n'a désormais plus qu'un seul état initial et final, l'état ω :

$$\mathcal{F}^*(X) = (E, \{\omega\}, \{\omega\}, T) \text{ avec } E = \{(u,v) \in \mathcal{A}^+ \times \mathcal{A}^+ \mid uv \in X\} \cup \{\omega\}$$

Seul 4 catégories de transitions sont possibles :

- $(u, av) \xrightarrow{a} (ua, v)$ pour $uav \in X, (u, v) \neq \omega, u, v \neq \epsilon$
- $\omega \xrightarrow{a} (a, v)$ pour $av \in X, v \neq \epsilon$,
- $(u, a) \xrightarrow{a} \omega$ pour $ua \in X, u \neq \epsilon$,
- $\omega \xrightarrow{a} \omega$ pour $a \in X$.

La figure (fig. 3.4) représente l'automate à pétales du code X = aa, ba, bb, baa, bba.

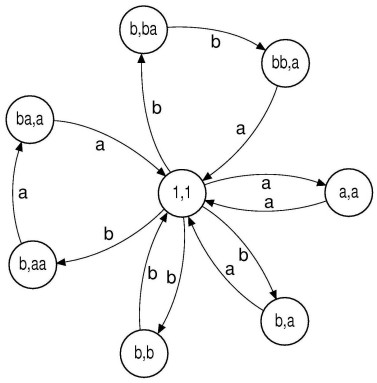

FIG. 3.4 – Automate à pétales de X

Chaque mot de $x \in X$ décrit donc un cycle dans l'automate à pétales $\omega \xrightarrow{x} \omega$.

Théorème

Soit X un sous-ensemble de $\mathcal{A}^+$. Les conditions suivantes sont équivalentes

(i) X est un code.

(ii) l'automate a pétales $\mathcal{F}^*(X)$ est non-ambigu.

Proposition

Soit $X \subset \mathcal{A}^+$ un code, pour tout mot $w \in \mathcal{A}^*$ et tout état $(u, v), (u', v')$ de l'automate $\mathcal{F}^*(X)$, les conditions suivantes sont équivalentes :

(i) il existe un chemin p dans $\mathcal{F}^*(X)$ tel que $p : (u, v) \xrightarrow{w} (u', v')$

(ii) On a soit $w \in vX^*u'$ soit $(uw = u'$ et $v = wv')$

Preuve (i) $\Rightarrow$ (ii) Si p est un chemin simple de $\mathcal{F}^*(X)$ qui ne passe pas par ω, alors $uw = u'$ et $v = wv'$. Sinon, le chemin p peut être décomposé de la façon suivante p : $(u, v) \xrightarrow{v} \omega \xrightarrow{x} \omega \xrightarrow{u'} (u', v')$ avec $w = vxu'$ et $x \in X^*$

La réciproque est évidente.

3.6 Codes Circulaires

3.6.1 Définition d'un code circulaire

Un ensemble $X \subset \mathcal{A}^*$ est un code circulaire si $\forall n, m \geq 1$, x_1, x_2, ..., $x_n \in X$, y_1, y_2, ..., $y_m \in X$ et $r \in \mathcal{A}^*$, $s \in \mathcal{A}^+$, l'égalité

$$sx_2, x_3, \ldots, x_n r = y_1, y_2, \ldots, y_m \text{ et } x_1 = rs$$

implique

$$n = m, r = \epsilon, x_i = y_i \text{ pour } i \in [1, n]$$

En d'autres termes, tout mot de $\mathcal{A}$ «écrit sur un cercle» a au plus une décomposition (factorisation) sur X (fig. 3.5). Les codes circulaires ont été introduits par Golomb et Gordon [GC65]. Certains codes circulaires possèdent de fortes propriétés de synchronisation (voir 3.7.).

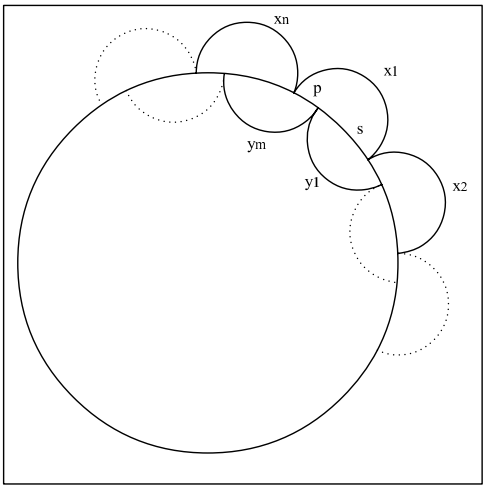

FIG. 3.5 – Deux factorisations circulaires d'un même mot

Tout sous-ensemble d'un code circulaire est aussi un code circulaire. Un code circulaire est un code. La réciproque n'est pas forcément vraie.

Exemple 1

Soit $X = \{aab, bba\}$ sur $\mathcal{A} = \{a, b\}$. X est un code circulaire. Si il existe une factorisation d'un mot w en mots de X (qui sont de la forme uuv avec $u, v \in \mathcal{A}$ et $u \neq v$), les mots de longueur 3 apparaissant en "décalé" sur un cercle sont de la forme vuu, uvu ou uuu et ne peuvent donc être des mots de X.

Exemple 2

Soit $X = \{aba, bab\}$ sur $\mathcal{A} = \{a, b\}$. X est un code mais X n'est pas circulaire. En effet, le mot $w = ababab$ peut être factorisé en mots de X de deux manières distinctes :

$$w = (aba)(bab) = a(bab)ab$$

Deux contraintes sur les mots appartenant à un code circulaire sont présentées.

Mots primitifs

Un mot $w \in \mathcal{A}^*$ est appelé *primitif* s'il n'est pas puissance d'un autre mot. Donc, w est primitif si et seulement si la condition $w = x^n$ implique $n = 1$. Nous dirons qu'un ensemble est primitif si tous ses éléments sont primitifs.

Tout mot d'un code circulaire X est primitif. Supposons que $w^n \in X$ avec $n \geq 2$, il alors possible d'écrire :

$$w(w^n)w^{(n-1)} = w^n w^n$$

D'après la définition, on a $w = \epsilon$ et donc une contradiction.

Mots conjugués

Deux mots w et w' sont appelés *conjugués* si il existe deux mots u et v tel que $w = uv$ et $w' = vu$ (fig. 3.6). w est appelé le conjugué de w'.

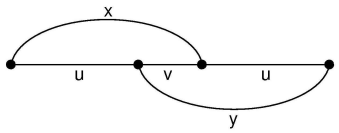

FIG. 3.6 – Deux factorisations circulaires d'un même mot

Deux mots conjugués peuvent être déduits l'un de l'autre par permutation circulaire. Soit γ une fonction de $\mathcal{A}^* \rightarrow \mathcal{A}^*$ définie par

$$\gamma(1) = 1 \text{ et } \gamma(lw) = wl \text{ pour } l \in \mathcal{A}, w \in \mathcal{A}^*$$

γ est une bijection de $\mathcal{A}^*$ dans $\mathcal{A}^*$. Deux mots w et w' sont conjugués si et seulement si il existe un entier $n \geq 0$ tel que

$$w = \gamma^n(w')$$

La relation de conjugaison est donc une relation d'équivalence. Elle permet de caractériser le sous-monoïde de $\mathcal{A}$ engendré par les codes circulaires. Une classe d'équivalence sur cette relation est appelée *classe de conjugaison* (certains auteurs emploient le terme de colliers [MR83], [Pir98]).

Un code circulaire ne peut contenir deux mots distincts conjugués. En effet, si $sr, rs \in X$ avec $s, r \in \mathcal{A}^+$ alors

$$s(rs)r = (sr)(sr)$$

X est un code circulaire implique que $r = 1$ ce qui est impossible.

Le nombre de classes de conjugaison pour des mots primitifs d'une longueur n sur un alphabet $\mathcal{A}$ de k lettres peut être calculé. Ce nombre est noté $l_n(k)$. Chaque mot de longueur n appartient à une unique classe de conjugaison. Chaque classe de conjugaison a $d = n/e$ éléments où e est l'exposant du mots. Il y a autant de classes dont les mots ont comme exposant n/e que de classses de mots primitifs de longueur $d = n/e$, d'où le résultat suivant

$$k^n = \sum_{d|n} d \cdot l_d(k)$$

où la somme est étendue sur tous les diviseurs d de n. Il est donc possible d'obtenir une expression explicite du nombre $l_n(k)$ en utilisant la méthode d'inversion de Moebius. On obtient alors

$$l_n(k) = \frac{1}{n} \sum_{d|n} \mu(d) k^{\frac{n}{d}}$$

où la somme est étendue sur tous les diviseurs d de k, et μ est la fonction de Moebius définie par :

$$\mu(d) = \begin{cases} 1 & \text{si } d = 1, \\ (-1)^i & \text{si } d = p_1 ... p_i, \text{ où } p_1, ..., p_r \text{ sont i nombres premiers distincts} \\ 0 & \text{sinon} \end{cases}$$

La table 3.1 contient le nombre de classes de conjugaison pour certains $l_n(k)$.

3.6.2 Monoïde et code circulaire

Nous allons caractériser les sous-monoïdes de $\mathcal{A}^*$ engendré par les codes circulaires.
Un sous-monoïde M de $\mathcal{A}^*$ est *pur* si pour tous $x \in \mathcal{A}^*$ et $n \geq 1$,

$$w^n \in M \Rightarrow w \in M$$

n	1	2	3	4	5	6	7	8	9	10	11	12
$l_n(2)$	2	1	2	3	6	9	18	30	56	99	186	335
$l_n(3)$	3	3	8	18	48	116						
$l_n(4)$	4	6	20									

TAB. 3.1 – Nombre de classes de conjugaison pour des codes contenant des mots de longueur n sur un alphabet de k lettres

Un sous-monoïde M de $\mathcal{A}^*$ est *très pur* si pour tout $u, v \in \mathcal{A}^*$,

$$uv, vu \in M \Rightarrow u, v \in M$$

Un monoïde très pur est toujours pur. La réciproque n'est pas toujours vraie.

Proposition

Un sous-monoïde M de $\mathcal{A}^*$ est très pur si et seulement si son ensemble minimal de générateurs est un code circulaire [BP85].

Preuve M est un sous-monoïde très pur. Montrons que M est stable. Soit $m, m', xm, m'x \in M$. Posons $u = x, v = mm'$ on alors $uv, vu \in M$. M est donc stable.

Comme M est stable, il est aussi libre, soit X sa base. Posons $u = s, v = x_2x_3 \dots x_np$ et uv et $vu \in M$. Par conséquent, $s \in M$. Comme $ps, x_2x_3 \dots x_np \in M$, la stabilité de M implique que $p \in M$. Comme $ps \in X$, il s'ensuit que $p = \epsilon$. Comme X est un code, cela implique que $n = m$ et $x_i = y_i$ pour $i \in [1, n]$.

Réciproquement, soit X un code circulaire et $M = X^*$. Pour montrer que M est très pur, considérons $u, v \in \mathcal{A}^*$ tel que $uv, vu \in M$. Soient $uv = x_1x_2 \dots x_n$ et $vu = y_1y_2 \dots y_m$ avec $x_i, y_i \in X$. Il existe un entier $i \in [1, n]$ tel que $u = x_1x_2 \dots x_{i-1}petv = sx_{i+1} \dots x_n$ avec $x_i = ps \in \mathcal{A}^*, s \in \mathcal{A}^+$. Le mot vu peut être écrit de deux façons $sx_{i+1} \dots x_nx_1x_2 \dots x_{i-1}p = y_1y_2 \dots y_m$ Comme X est un code circulaire, il s'ensuit que $p = \epsilon$ et $s = y_1$.

Par conséquent, $u, v \in M$. M est donc très pur.

3.6.3 Test de circularité

Proposition

Soit $X \subset \mathcal{A}^+$ un code et l'automate à pétales $\mathcal{F}^*(X)$. Les différentes conditions sont équivalentes

(i) X est un code circulaire.

(ii) Soit l'automate $\mathcal{F}^* = (Q, \{\epsilon\}, \{\epsilon\}, T)$ et $p, q \in Q$. Pour tout $w \in \mathcal{A}^+$, $(p, w, p), (q, w, q) \in T^* \Rightarrow p = q$

Preuve

(i) $\rightarrow$ (ii). Soit $w \in \mathcal{A}^+$ et deux états de l'automate à pétales $\mathcal{F}^*(X)$. Comme $w \neq \epsilon$, la proposition implique que $w \in vX^*u$ et $w \in v'X^*u'$. Chacun des cycles $c : p \xrightarrow{w} p$ et $c' : q \xrightarrow{w} q$ passe par l'état ω. On considère que $v \leq v'$. Soit $z, t \in \mathcal{A}^*$ les mots tels que $v' = vz$ et $w = v't = vzt$. Les chemins c et c' peuvent donc être factorisés de la façon suivante

$$c : p \xrightarrow{v} \omega \xrightarrow{z} r \xrightarrow{t} p$$

$$c' : q \xrightarrow{v} s \xrightarrow{z} \omega \xrightarrow{t} q$$

Donc les chemins suivants existent

$$d : \omega \xrightarrow{z} r \xrightarrow{t} p \xrightarrow{v} \omega \text{ et } d' : \omega \xrightarrow{t} q \xrightarrow{v} s \xrightarrow{z} \omega$$

L'existence des chemins d et d' implique que $ztv, tvz \in X^*$. Comme X^* est très pur alors $z, tv \in X^*$ ce qui implique qu'il existe aussi un chemin $e : \omega \xrightarrow{z} \omega \xrightarrow{tv} \omega$. Comme $\mathcal{F}^*$ est un automate à pétales, il est non ambigu, $d = e$ et donc $r = \omega$. On a alors $f : \omega \xrightarrow{t} p \xrightarrow{vz} \omega$. La comparaison de f à d' implique que $p = q$.

(ii) $\Rightarrow$ (i). Soit $u, v \in \mathcal{A}^*$ tel que $uv, vu \in X^*$. Il existe donc deux chemins $c : \omega \xrightarrow{u} p \xrightarrow{v} \omega$ et $c' : \omega \xrightarrow{v} q \xrightarrow{u} \omega$. On a donc deux cycles étiquetés du même mot uv l'un dont l'origine est ω et l'autre dont l'origine est q. D'après (ii), on a donc $q = \omega$ et donc $u, v \in X^*$. On a $uv, vu \in X^* \Rightarrow u, v \in X^*$ donc X^* est un monoïde très pur et X est un code circulaire.

Pour prouver qu'un ensemble X est un code circulaire, il suffit de prouver qu'il n'existe pas deux cycles distincts étiquetés par le même mots.

3.7 Classes de codes améliorant le décodage

Le décodage d'un mot généré par un code ou par un code circulaire nécessite généralement de commencer au début du mot et de se dérouler de façon séquentielle. Toute erreur dans le message ou perte de la phase de lecture provoque donc l'impossibilité de poursuivre le décodage. Des codes plus robustes ont donc été construits.

3.7.1 Codes synchronisants

Définition

Un code X est *synchronisant* s'il existe $x, y \in X^*$ tels que pour tout $u, v \in \mathcal{A}^*$ on a :

$$uxyv \in X^* \Rightarrow ux, yv \in X^*$$

La paire (x, y) joue un rôle particulier dans le décodage. Elle est dite synchronisante pour X. Ces codes permettent d'effectuer en partie le décodage même si une erreur de transmission se produit. Ainsi, lors de la réception du mot $uxyv$ tel que (x, y) est une paire synchronisante. Lorsqu'une erreur se produit dans u, il reste toujours possible de poursuivre le décodage de yv.

Remarque : Si un code X admet (ϵ, ϵ) comme paire synchronisante, alors $X \subset \mathcal{A}$. En effet, soit $x \in X$ et l la première lettre de x. Comme X est un code, $x \neq \epsilon$. Posons $x' = l^{-1} \cdot x$. Alors $l \cdot \epsilon \cdot \epsilon x' = x \in X^*$ donc, comme (ϵ, ϵ) est une paire synchronisante, on obtient $l \in X^*$ et $x' \in X^*$. Puisque X est un code, on a ainsi $x' = \epsilon$, soit $x = l$.

Il est possible d'établir une relation entre code circulaire et code synchronisant. Commençons par introduire la notion de code coupant.

Un code X est *coupant* s'il existe un mot sur $\mathcal{A}^*$ qui n'est pas facteur de $X(\mathcal{A}^* \neq F(X))$.

On a alors d'après [BP85]
Un code circulaire coupant est synchronisant.

Les codes synchronisants nécessitent l'insertion de mots particuliers de synchronisation dans le message. Il existe une classe de code offrant des possibilités de synchronisation plus générales, ce sont les codes uniformément synchronisants [GG65].

3.7.2 Codes uniformément synchronisants

Définition

Un code X est *uniformément synchronisant* (ou à délai de synchronisation borné) s'il existe un entier $\sigma \geq 0$ tel que pour tout $x, y \in X^\sigma$ et $u, v \in \mathcal{A}^*$ on a

$$uxyv \in X^* \Rightarrow ux, yv \in X^*$$

En d'autres termes, toute paire de $X^\sigma \times X^\sigma$ est synchronisante. Le plus petit entier σ vérifiant la formule ci-dessus est le délai de synchronisation de X.

La synchronisation s'effectue donc pour tout mot suffisamment long.

Tout code uniformément synchronisant est un code circulaire.

La réciproque n'est pas forcément vraie mais il est possible d'établir que :

Théorème

Soit X un code finie, les conditions suivantes sont équivalentes [BP85] :

(i) X est un code circulaire

(ii) X est uniformément synchronisant

3.7.3 Code à délai d'interprétation fini

Une autre classe de code se base sur la notion d'interprétation (2.4.) afin de permettre un décodage à partir d'une fraction de code, ce sont les codes à délai d'interprétation fini [Gue00].

Définition

Soit X un ensemble sur $\mathcal{A}^+$. X est un code à *délai d'interprétation fini* s'il existe $n \geq 1$ tel que pour tout $\alpha \in P(X)$, $\beta \in S(X)$, $(\alpha, \beta) \notin X^* \times X^*$, on a :

$$\beta X^* \alpha \cap X^n = \emptyset$$

Le délai d'interprétation est le plus petit entier vérifiant la condition précédente.
Un délai d'interprétation n permet de mettre en évidence que les mots de X^n n'admettent que des interprétations quasi triviales.

Un ensemble X est une partie adjacente si et seulement si on a

$$X \cap \overline{P(X)}.X^+ = \emptyset \text{ et } X \cap X^+.\overline{S(X)} = \emptyset$$

Un *code adjacent* est une partie adjacente de $\mathcal{A}^*$ ne contenant pas le mot vide.

Remarque

Tout code uniforme est adjacent.

Théorème [Gue01]

Soit X un code uniformément synchronisant. Les deux propriétés sont équivalentes :

(i) X est adjacent.

(ii) X est un code à délai d'interprétation fini.

La figure 3.7 propose une représentation des relations entre les différentes classes de codes.

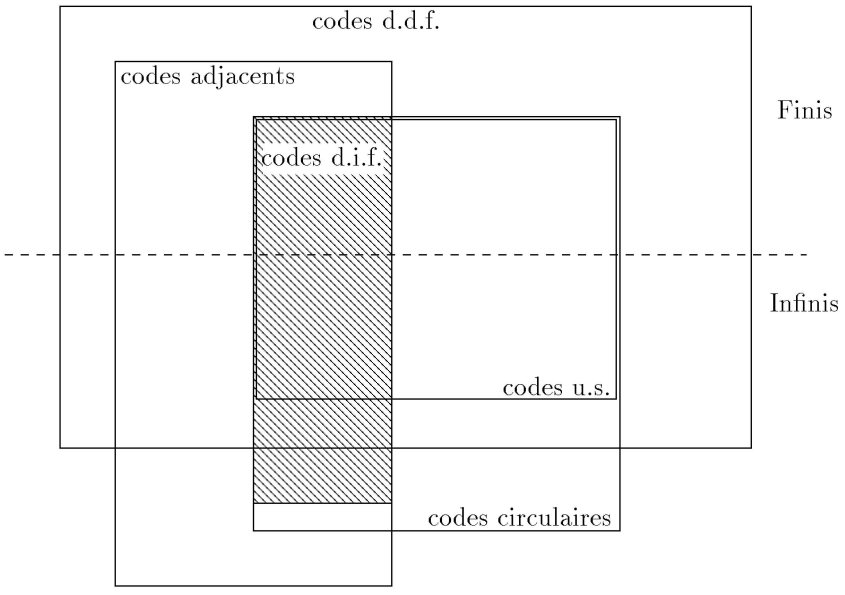

FIG. 3.7 – Relations entre classes de codes [Gue01]

Nous notons F_{code} (resp. F_{circ} , F_{ddf} , F_{sync} , F_{us}) l'ensemble de tous les codes (resp. codes circulaires, codes à délai de déchiffrage fini, codes synchronisants, codes uniformément synchronisants).

Pour tout entier $d \geq 0$, nous notons F_{ddf}^d (resp. F_{us}^d) l'ensemble de tous les codes à délai de déchiffrage d (resp. codes uniformément synchronisants de délai de synchronisation d).

3.7.4 Codes comma free

Les codes comma free peuvent être considérés comme la famille de codes la plus restrictive : pour tous mots w construit par la concaténation de mots d'un comma free code, les mots du code comma free apparaîtront uniquement dans la X-factorisation de w.

Définition

Un ensemble X de $\mathcal{A}^*$ est *comma free* si et seulement si

$$X^2 \cap \mathcal{A}^+ X \mathcal{A}^+ = \emptyset$$

Les codes comma free sont les codes avec le déchiffrage le plus simple. Pour tout mot $w \in X^*$, si un facteur du mot peut être identifié comme appartenant à X, alors ce facteur est un terme unique de la X-facteurisation de w. Les codes comma free sont circulaires.

3.8 Maximalité

3.8.1 Code maximal

Définition

Un code X est *maximal* si pour tout code $\mathcal{Y}$ tel que $X \subset \mathcal{Y}$ on a $\mathcal{Y} = X$. X ne peut donc pas être contenu dans un code plus grand.

Il est possible de montrer grâce au lemme de Zorn que tout code est inclus dans un code maximal. Les codes maximaux ont alors une importance particulière. La structure des codes maximaux renseigne sur la structure des codes en général. Les codes maximaux utilisent en quelque sorte toute la capacité du canal de transmission.

La maximalité renseigne sur la structure d'un code et en particulier sur son pouvoir générateur. Caractérisons la structure des codes.

Un code qui n'est pas coupant est *dense*.

Un code X est *complet* si $F(X^*) = \mathcal{A}^*$, c'est-à-dire X^* est dense.

Il est possible dans certains cas de lier la complétude d'un code à la maximalité.

Théorème

Tout code maximal est complet [Niv66].

Théorème

Tout code coupant et complet est aussi maximal [NS01].

Remarque

Dans le cas des codes finis, il n'est pas possible de dire en général si un code fini est inclus dans un code fini maximal.

Par exemple, $\{a^5, ba^2, ab, b\}$ n'est inclus dans aucun code maximal fini [Mar67]. Décider si un code fini est inclus dans un code fini maximal reste un problème ouvert.

Concernant la maximalité des codes circulaires :

Proposition [BP85]

Soit $X \subset \mathcal{A}^+$ un code circulaire. Si X est maximal en tant que code circulaire alors X est complet.

Théorème

Soit X un code circulaire coupant. Les propositions suivantes sont équivalentes :

(i) X est complet

(ii) X est un code maximal

(iii) X est maximal en tant que code circulaire

Par conséquent, le seul code circulaire maximal fini est l'alphabet $\mathcal{A}$.

3.8.2 Complétion

Différentes méthodes de complétions existent. Une méthode générale permet de compléter un code [ER85]. Cette méthode utilise la notion de marqueur :

Théorème

Soit X un code coupant non maximal.

(i) Il existe un mot sans bord y tel que $y \notin F(X^*)$

(ii) Soient U $= \{\{\mathcal{A}^* \setminus X^*\} \setminus \mathcal{A}^* \{y\} \mathcal{A}^*\}$

Alors $Y = X \cup y(Uy)^*$ est un code coupant maximal. De plus, si X est rationnel alors Y l'est aussi. Dans cet algorithme, le mot y est utilisé comme "marqueur" : les mots ajoutés à X commencent et se terminent tous par y.
Si X est un code circulaire coupant, la même construction conduit à un code Y qui est circulaire, coupant et maximal dans F_{circ}.

Une méthode de complétion existe aussi pour compléter les codes comma free réguliers [Lam03].

3.9 Un cas pratique : Le code circulaire $\mathcal{X}$

L'ensemble $\mathcal{X}$ = {AAC, AAT, ACC, ATC, ATT, CAG, CTC, CTG, GAA, GAC, GAG, GAT, GCC, GGC, GGT, GTA, GTC, GTT, TAC, TTC} est le résultat d'une étude statistique de populations de gène d'eucaryote et procaryote (2.4). $\mathcal{X} \subset \mathcal{A}^*$ avec $\mathcal{A} = \{A, C, G, T\}$, il contient uniquement des mots de longueur 3. Par conséquent, $\mathcal{X}$ un code uniforme.

Pour vérifier que $\mathcal{X}$ est un code circulaire, l'automate à pétales λ déterministe associé au code $\mathcal{X}$ est construit (fig 3.8). Conformément à 3.5.3, il suffit de vérifier que l'automate à pétales λ ne contient pas de cycles distincts étiquetés par le même mot.

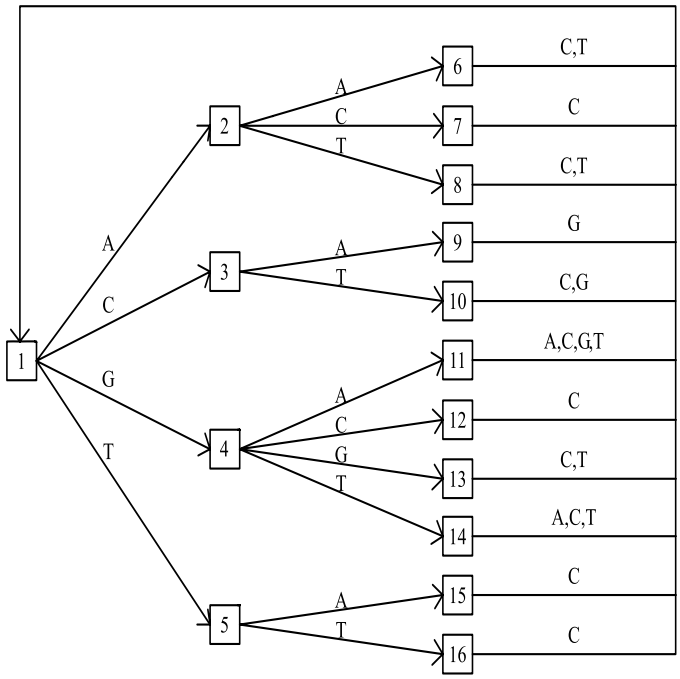

FIG. 3.8 – Automate à pétales de $\mathcal{X}$

Tous les pétales de l'automate λ ont une longueur de 3 lettres. Il est donc possible d'attribuer une profondeur $k \in [0, 2]$ à chaque états, $k = 0$ correspondant à l'état ω. L'ensemble $\mathcal{X}$ étant un code uniforme, il ne peut y avoir de cycles distincts dont les origines sont des états de même profondeur. Les cycles doivent donc avoir des origines correspondant à des états de profondeur différentes.

Un simple parcours de l'automate λ et une recherche exhaustive des cycles montre que λ ne contient pas de cycles différents et étiquetés par un même mot et donc que $\mathcal{X}$ est un code circulaire.

L'ensemble $\mathcal{X}$ est un code fini et circulaire ce qui implique d'après 3.7.2 que $\mathcal{X}$ est un code uniformément synchronisant. Comme $\mathcal{X}$ est uniforme, il est donc aussi adjacent (3.7.3) et donc aussi à délai d'interprétation fini. Les plus longs mots pouvant appartenir à deux phases ont 12 lettres. Le délai de synchronisation est de 5 et le délai d'interprétation de 4.

D'après 3.8.1, le seul code circulaire maximal fini est l'alphabet $\mathcal{A}$. Par conséquent, $\mathcal{X}$ n'est pas maximal. Cependant d'après 2.6.1, un code uniforme de mots de longueur 3 sur un alphabet de 4 lettres a 20 classes de conjugaison. Un code circulaire peut contenir au plus un mot par classe de conjugaison et il contiendra au maximum 20 mots. $\mathcal{X}$ est donc le plus grand code circulaire uniforme possible. L'ensemble $\mathcal{X}$ est le plus grand code circulaire uniforme de mots 3 lettres sur un alphabet de 4 lettres.

Dans la suite, nous continuons à décrire $\mathcal{X}$ comme étant un code circulaire bien qu'il soit possible de le caractériser par d'autres classes de codes. Les codes circulaires considérés comme maximaux seront les plus grands codes circulaires uniformes (mots de longueur 3 sur un alphabet de 4 lettres), c'est à dire les codes circulaires contenant 20 mots.

Chapitre 4

Etude de phases préférentielles dans les gènes

4.1 Génomique

Depuis plusieurs années, un changement d'échelle s'est opéré dans la recherche en génétique. Un grand nombre de données génétiques sont désormais connues et disponibles. De plus en plus d'études partent désormais directement de l'information génétique. Des hypothèses sur les fonctions des gènes peuvent être issues de recherches par ordinateur (*in silico*) et non plus d'observations expérimentales. Cette recherche à partir des données est possible même sans de fortes connaissances contextuelles préalables. Elle permet par inférences de proposer des hypothèses qui peuvent ensuite être confrontées au réelles par l'expérience.

La génomique reste une discipline scientifique jeune. On appelle génome l'ensemble des informations génétiques d'un individu. L'information génétique est généralement conservée sous la forme d'ADN ou d'ARN. L'étape de séquençage consiste à déterminer la séquence de nucléotides d'un génome. Le premier génome à ADN entièrement séquencé a été celui du bactériophage *ϕX174* en 1978 [San78]. Le premier procaryote séquencé est l'*Haemophilus influenzae* en 1995 [Fle95]. La première archaea séquencée a été le *Methanococcus jannaschii* [Bul96]. Le premier génome d'eucaryote disponible est celui d'une levure, le *Saccharomyces Cerevisiae* [Duj96]. Les levures sont un modèle fortement étudié chez les eucaryotes. Ce séquençage permet donc de nombreuses études comparatives. Chez les procaryotes, le modèle le plus important est certainement l'*Escherichia coli*. Un grand nombre d'études (génétiques, métaboliques, épidémiologiques, etc.) ont été consacré à cette bactérie. Son génome a été séquencé pour la première fois en 1997 [Bla97].

Grâce à l'amélioration des techniques de séquençage et à l'intérêt croissant pour ce type de projet, le nombre de génomes disponibles n'a cessé d'augmenter. Au moment de notre travail, les données génétiques de 16 archaea et de plus de 170 bactéries (voir tableau 4.3) étaient accessibles. Dans un premier temps, il a été décidé que notre étude porterait uniquement sur les génomes de procaryotes. Plusieurs raisons ont justifié ce choix. La principale est que la quantité de génomes disponibles est nettement plus importante chez les procaryotes. Des études comparatives sont alors possibles. En particulier, pour certaines espèces de bactéries, les génomes de plusieurs souches différentes ont été séquencés (par exemple, *Helicobacter pilori 226695* et *Helicobacter pilori j99*). De plus les gènes de procaryotes ont généralement une structure plus simple. Les gènes de procaryotes n'ont pratiquement jamais d'intron (séquences de nucléotides supprimés lors d'une étape appelée épissage, les introns ne sont donc pas codant pour les protéines). Tous les génomes de

procaryote analysés sont répertoriés dans le tableau 4.3, les bactéries correspondent aux lignes de C_0 à C_{71} tandis que les archaea se situent aux lignes C_{72} à C_{85} (à l'exception de l'archaea *Halobacterium sp. NCR-1* situé à la ligne C_0).

4.2 Code $\mathcal{C}^3$

Le premier code circulaire identifié (voir 2.5) est l'ensemble $\mathcal{X}_0$ (EUK_PRO) = {AAC, AAT, ACC, ATC, ATT, CAG, CTC, CTG, GAA, GAC, GAG, GAT, GCC, GGC, GGT, GTA, GTC, GTT, TAC, TTC} . Un code circulaire est un ensemble de mots sur un alphabet tel que tous mot écrit sur un cercle (la lettre suivant la dernière lettre d'un mot est la première lettre du mot) a au plus une décomposition en mots du code. La phase de construction d'un mot construit par la concaténation de mots d'un code circulaire peut être retrouvée après la lecture, n'importe où dans le mot, d'un petit nombre de nucléotide. Cette suite de nucléotides est appelée la fenêtre d'un code circulaire et est fonction du code. La notion de fenêtre est similaire à celle de délai (cf. la section 3.7.3), à la différence que le délai est exprimé en nombre de mots tandis que la fenêtre est exprimée en nombre de lettres. La longueur de la fenêtre d'un code circulaire composé de trinucléotide est au maximum de 13 lettres.

L'ensemble $\mathcal{X}_0$ est un code circulaire. Les ensembles $\mathcal{X}_1$ (resp. $\mathcal{X}_2$) obtenus par permutation circulaire gauche de 1 lettre (resp. 2 lettres) de tous les mots de $\mathcal{X}_0$ est aussi un code circulaire. Un code circulaire dont les ensembles permutés sont aussi des codes circulaires sera appelé code $\mathcal{C}^3$. Tous les codes circulaires ne sont pas forcément $\mathcal{C}^3$. Par exemple, l'ensemble X=GGT, TTG est un code circulaire (voir 3.6.1, exemple 1) mais son ensemble permuté gauche X_G=GTG, TGT n'est pas un code circulaire (voir 3.6.1, exemple 2).

D'après la section 3.6.1, une condition nécessaire mais non suffisante pour qu'un code soit circulaire est l'absence de mots primitifs et de mots permutés distincts. Par conséquent, les 60 trinucléotides, sans AAA, CCC, GGG et TTT, peuvent être groupés en 20 classes de 3 trinucléotides invariant par permutation. Pour chaque trinucléotide, nous nous intéressons simultanément à sa fréquence d'occurrence dans chacune des 3 phases potentielles des séquences, technique réalisé avec la méthode FTF (cf. 2.4), mais aussi aux fréquences d'occurrence d'un trinucléotide par rapport aux fréquences des 2 autres trinucléotides de son groupe de permutation. La méthode, prenant en compte à la fois les trinucléotides par phase préférentielle et les groupes de permutation sera notée méthode FPTF pour Frame Permutated Trinucleotide Frequency.

4.3 Méthode FPTF

La méthode FPTF est donc une fonction statistique s'appuyant sur ces deux paramètres, permettant ainsi d'associer à chaque trinucléotide permuté sa phase préférentielle.

Une séquence de nucléotides codant pour un acide aminé à 3 phases de lectures potentielles (dans le cas où la lecture sur un seul brin de l'ADN est uniquement considérée), la phase $p = 0$ correspondant au cadre de lecture utilisé lors de transcription/traduction, établi par le codon initiateur et les phase p=1 (resp. p=2) qui correspondent à la phase 0 décalée de 1 (resp. 2) nucléotides dans le sens de lecture de la séquence.

Les 60 trinucléotides non-identiques ($\neq$ AAA, CCC, GGG et TTT) peuvent être groupés en 20 ensembles de 3 nucléotides identiques par permutation circulaire. Ces ensembles seront notés S_j avec $j \in \{0, \ldots, 19\}$: S_0 ={AAC, ACA, CAA}, S_1 ={AAG, AGA, GAA}, ..., S_{19} ={GTT, TTG, TGT}. Par convention les ensembles sont indicés selon l'ordre lexicographique de leur plus petit mot. Chaque ensemble S_j contient les mots w_i avec $i \in \{0, 1, 2\}$ tel que w_0 est le plus petit mot de S_j suivant l'ordre lexicographique et $w_1 = P(w_0)$, $w_2 = P(P(w_0))$.

Exemple

$$S_0 = \{w_0\text{=AAC}, w_1\text{=ACA}, w_2\text{=CAA}\}$$

Le trinucléotide w lu dans la phase p sera noté w^p. Le trinucléotide w_i, $i \in \{0, 1, 2\}$ d'un ensemble S_j lu dans la phase $p \in \{0, 1, 2\}$, est noté w_i^p. Un groupe G_j, $j \in \{0, \ldots, 19\}$, associé à l'ensemble S_j, $j \in \{0, \ldots, 19\}$ contient les $3 \times 3 = 9$ mots w_i^p

Exemple

G_0 est associé à S_0,

$$G_0 = \{\text{AAC}^0, \text{ACA}^1, \text{CAA}^2, \text{AAC}^1, \text{ACA}^2, \text{CAA}^0, \text{AAC}^2, \text{ACA}^0, \text{CAA}^1\}$$

La fréquence d'occurrence de chaque trinucléotide w_i^p, i, $p \in \{0, 1, 2\}$ de chaque groupe G_j, $j \in \{0, \ldots, 19\}$, est comparée aux fréquences de $w_i^{p'}$ et $w_i^{p''}$ avec $p \neq p' \neq p''$ et aux fréquences de $w_{i'}^p$ et $w_{i''}^p$ avec $i \neq i' \neq i''$. La fréquence d'un trinucléotide dans une phase p est ainsi comparée aux fréquences de ses trinucléotides permutés dans la même phase p et la fréquence d'un trinucléotide w dans une phase est comparée à la fréquence du même

trinucléotide w dans les autres phases de lectures potentielles. Il est donc nécessaire de calculer deux composantes.

Soit $\sigma(w_i^p)$ la fréquence d'occurrence observée d'un trinucléotide dans la phase p d'un ensemble de gènes. Pour un groupe G_j, la fonction $\mathcal{Q}(w_i^p)$ estime la contribution de w_i^p à la phase p par rapport à celle de ses trinucléotides équivalent par permutation

$$\mathcal{Q}(w_i^p) = \frac{\sigma(w_i^p)}{\sum_{r=0}^{2} \sigma(w_r^p)}$$

De la même façon, pour un groupe G, la fonction $\mathcal{P}(w_i^p)$ le rapport de la fréquence d'occurrence d'un trinucléotide w_i^p dans une phase sur celles du même mot w_i dans toutes les phases

$$\mathcal{P}(w_i^p) = \frac{\sigma(w_i^p)}{\sum_{r=0}^{2} \sigma(w_i^r)}$$

Remarque

Les dénominateurs de $\mathcal{P}(w_i^p)$ et de $\mathcal{Q}(w_i^p)$ doivent être supérieurs à zéro. En pratique, un grand nombre de gènes est étudié et par conséquent la somme des $\sigma(w_i^p)$ est toujours positive.

Un trinucléotide w_i^p apparaissant avec une fréquence élevée dans une phase p par rapport à celles des autres phases peut avoir un nombre d'occurrence faible dans cette phase par rapport à celle des ses nucléotides permutés. Etant donné que l'on accorde une importance équivalente à ces deux critères, une fonction $\mathcal{M}(w_i^p)$ sera définie comme étant la moyenne de $\mathcal{Q}(w_i^p)$ et $\mathcal{P}(w_i^p)$

$$\mathcal{M}(w_i^p) = \tfrac{1}{2}(\mathcal{P}(w_i^p) + \mathcal{Q}(w_i^p))$$

La valeur renvoyée par $\mathcal{M}(w_i^p)$ est comprise entre 0 et 1.

$\mathcal{M}(w_i^p)$ évalue simultanément la contribution de w_i^p à sa phase et à son groupe de permutation. Le trinucléotide w_i^p avec la valeur la plus forte sera associé de façon préférentielle à la phase p. Une valeur très forte de $\mathcal{M}(w_i^p)$ (c'est à dire proche de 1) signifie que le trinucléotide w_i^p est associé de façon fortement préférentielle à p, c'est à dire que les deux trinucléotides permutés w_i^p n'apparaissent pas de manière significative dans la phase p et que la fréquence d'occurrence de w_i^p est faible dans les deux autres phases p' et p'' $(p \neq p' \neq p'')$.

Remarque

La fonction $\mathcal{M}(w_i^p)$ peut être écrit de façon plus générale en donnant un coefficient à $\mathcal{P}(w_i^p)$ et à $\mathcal{Q}(w_i^p)$

$$\mathcal{M}(w_i^p) = (\alpha\mathcal{P}(w_i^p) + \beta\mathcal{Q}(w_i^p)) \text{ avec } 0 \leq \alpha, \beta \leq 1 et \alpha + \beta = 1$$

Les valeurs de α et β dépendront de l'importance accordée au poids d'un nucléotide dans sa phase où dans son groupe de permutation. Par exemple, pour la valeur $\beta = 0$, un mot w_i^p est uniquement comparé aux fréquences de w_i dans les autres phases. La méthode FPTF revient alors à la méthode FTF (chapitre 2.4). Cependant, seul les résultats obtenus à partir de la fonction $\mathcal{M}(w_i^p)$ avec $\alpha = \frac{1}{2}$, $\beta = \frac{1}{2}$ seront présents ici.

L'étape suivante de la méthode FPTF consiste à sélectionner l'ensemble $\mathcal{S}$ de 3 trinucléotides w_i^p d'un groupe G_j , $j \in \{0, \ldots, 19\}$ en fonction des valeurs de $\mathcal{M}(w_i^p)$. Un groupe G a 9 trinucléotides. Par conséquent, il y a $\binom{9}{3} = 84$ ensembles possibles de 3 valeurs. Ces 84 ensembles $\mathcal{S}$ sont définis de la façon suivante : $\{\{w_0^0, w_0^1, w_0^2\}, \ldots, \{w_0^0, w_0^2, w_1^0\}, \ldots, \{w_0^0, w_1^0, w_1^1\}, \ldots, \{w_0^0, w_1^1, w_1^2\}, \ldots, \{w_0^0, w_1^2, w_2^0\}, \ldots, \{w_0^1, w_0^2, w_1^0\}, \ldots, \{w_0^2, w_1^0, w_1^1\}, \ldots, \{w_1^0, w_1^1, w_1^2\}, \ldots, \{w_1^1, w_1^2, w_2^0\}$, $\ldots, \{w_1^2, w_2^0, w_2^1\}, \ldots, \{w_2^0, w_2^1, w_2^2\}$ $\} = \{\mathcal{S}_0, \ldots, \mathcal{S}_{84}$ $\}$.

Les ensembles sont numérotés $\mathcal{S}_N$ tel que pour $w_{i_0}^{p_0}, w_{i_1}^{p_1}, w_{i_2}^{p_2}\}$, $N = \frac{1}{6}$ $I_0(I_0^2$ - $24I_0 + 191)$ - $\frac{1}{2}I_1(I_1 - 17)$ + I_2 - 45 avec $I_0 = 3 * i_0 + p_0 + 1$, $I_1 = 3i_1 + p_1 + 1$, $I_2 = 3i_2 + p_2 + 1$, $i_0, i_1, i_2, p_0, p_1, p_2 \in \{0, 1, 2\}$ et $I_0 < I_1 < I_2$.

Parmi ces 84 ensembles, 3 ensembles sont plus particulièrement intéressants : les ensembles $\mathcal{S}_{21} = \{w_0^0, w_1^1, w_2^2\}$, $\mathcal{S}_{43} = \{w_0^1, w_1^2, w_2^0\}$ et $\mathcal{S}_{52} = \{w_0^2, w_1^0, w_2^1\}$. En effet, ces ensembles ont la particularité d'associer chaque trinucléotide à une phase de façon à ce que, si w_i est associé à la phase p, alors w_{i+1} est associé à $p + 1(mod 3)$. Ainsi lorsqu'un trinucléotide est associé à une phase préférentielle, le trinucléotide obtenu par permutation circulaire gauche est associé à la phase suivante. Ainsi, l'association d'un trinucléotide à une phase permet de déduire par permutation les phases associées préférentiellement aux autres trinucléotides du même groupe de permutation.

Afin d'évaluer la significativité des ensembles $\mathcal{S} = \{w_{l_0}^{p_0}, w_{l_1}^{p_1}, w_{l_2}^{p_2}\}$ (et donc de la correspondance entre trinucléotides et phases préférentielles), la fonction $\mathcal{F}(\mathcal{S})$ est définie comme étant la moyenne de la fonction $\mathcal{M}(w_i^p)$ appliquée à chacun des 3 mots $w_{l_0}^{p_0}$, $w_{l_1}^{p_1}$ et $w_{l_2}^{p_2}$

$$\mathcal{F}(\mathcal{S}) = \mathcal{F}(\{w_{l_0}^{p_0}, w_{l_1}^{p_1}, w_{l_2}^{p_2}\}) = \tfrac{1}{3}(\mathcal{M}(w_{l_0}^{p_0}) + \mathcal{M}(w_{l_1}^{p_1}) + \mathcal{M}(w_{l_2}^{p_2}))$$

Propriété

Un groupe G contient 9 trinucléotides w_i^p avec $i, p \in \{0, 1, 2\}$. Les ensembles $\mathcal{S}$ contiennent 3 trinucléotides w_i^p. G peut donc être partitionné et par conséquent, $\forall x, \exists y, z \in \{0, \ldots, 83\}$ de tel sorte que

$$\mathrm{G} = \mathcal{S}_x \cup \mathcal{S}_y \cup \mathcal{S}_z$$

Cette partition conduit ainsi à la relation suivante

$$\mathcal{F}(\mathcal{S}_x) + \mathcal{F}(\mathcal{S}_y) + \mathcal{F}(\mathcal{S}_z) = 1$$

Et par conséquent, le résultat de $\mathcal{F}(\mathcal{S})$ est compris entre 0 et 1. Par ailleurs,

$$\mathcal{F}_{max} = MAX\{\mathcal{F}(\mathcal{S}_x), \mathcal{F}(\mathcal{S}_y), \mathcal{F}(\mathcal{S}_z)\} \Rightarrow \tfrac{1}{3} \leq \mathcal{F}_{max} \leq 1$$

Propriété

Lorsque les 9 fréquences d'occurrences $\sigma(w_i^p)$ d'un groupe G associées à un ensemble S_j, $j \in \{0, \ldots, 19\}$, ont des valeurs similaires, c'est à dire $\sigma(w_i^p) = \sigma(w_{i'}^{p'}), \forall i, i', p, p' \in \{0, 1, 2\}$, les 84 fonctions $\mathcal{F}(\mathcal{S})$ ont des valeurs proches et égales à $\mathcal{F}(\mathcal{S}_k) = \frac{1}{3}$ $\forall k \in \{0, \ldots, 83\}$.

L'ensemble $\mathcal{S}_{max}$ est celui ayant la plus haute valeur pour la fonction $\mathcal{F}(\mathcal{S})$)

$$\mathcal{S}_{max} = \mathcal{S}_k \text{ tel que } \mathcal{F}(\mathcal{S}_k) = MAX_{k=0}^{83}\{\mathcal{F}\{\mathcal{S}_k)\}$$

Les 84 ensembles $\mathcal{S}$ sont ordonnés en fonction du résultat de $\mathcal{F}(\mathcal{S})$. Un rang leur est attribué, $\mathcal{S}_{max}$ ayant le rang 1.

L'ensemble $\mathcal{S}_{max}$ est déterminé pour chaque groupe G_j, $j \in \{0, \ldots, 19\}$, de chaque ensemble de séquences étudiées. Dans la majorité des cas, $\mathcal{S}_{max}$ est un des trois ensemble intéressant $\mathcal{S}_{21}$, $\mathcal{S}_{43}$ et $\mathcal{S}_{52}$. Si ce n'est pas le cas, on cherche à déterminer lequel de ces 3 ensembles correspond à l'association de trinucléotides à une phase préférentielle la plus significative

$$\mathcal{S}_{pref} = \mathcal{S}_k \text{ tel que } \mathcal{F}(\mathcal{S}_k) = MAX_{k=21,43,52}\{\mathcal{F}(\mathcal{S}_k)\}$$

$\mathcal{S}_{pref} \in \{\mathcal{S}_{21}, \mathcal{S}_{43}, \mathcal{S}_{52}\}$. $\mathcal{S}_{max}$ est utilisé comme référence. La différence de valeur entre $\mathcal{F}(\mathcal{S}_{pref})$ et $\mathcal{F}(\mathcal{S}_{max})$ ainsi que le rang $\mathcal{F}(\mathcal{S}_{pref})$ permettent d'évaluer l'ensemble sélectionné.

Pour un ensemble de séquences $\mathcal{G}$, S_{pref} est calculé pour chacun des 20 groupes G. L'ensemble S associé à un groupe particulier G_i est noté S^i_{pref} avec $i \in \{0, \ldots, 19\}$. Les différents ensembles obtenus sont donc S^0_{pref}, S^1_{pref}, ..., S^{19}_{pref}. Chacun des ensembles S^i_{pref} contient les trinucléotides d'un même groupe de permutation associés chacun à une phase de lecture différente. La méthode FPTP permet donc l'identification de 20 ensembles préférentiels $\mathcal{S}_{pref}$ de 3 trinucléotides pour chaque ensemble de séquences étudié de telle sorte que 3 trinucléotides permutés sont assignés à 3 phases de lecture des gènes. L'ensemble des trinucléotides associés à la phase 0 d'un ensemble de séquences $\mathcal{G}$ (resp. 1 et 2) forme l'ensemble $\mathcal{X}_0(\mathcal{G})$ (resp. $\mathrm{X}_1(\mathcal{G})$, $\mathrm{X}_2(\mathcal{G})$). Les propriétés des ensembles $\mathrm{X}_0(\mathcal{G})$, $\mathrm{X}_1(\mathcal{G})$ et $\mathrm{X}_2(\mathcal{G})$ obtenus sont ensuite étudiées. En particulier, il est possible de vérifier s'ils correspondent à des codes circulaires. La méthode FTPF a été appliquée à différents ensembles de séquences.

4.4 Etude massive de génomes de procaryotes

4.4.1 Application de la méthode FPTF

Une recherche des ensembles de trinucléotides associés aux différentes phases préférentielles d'un grand nombre de séquences nucléiques a été effectuée. En particulier, la méthode FPTF a été appliquée à la totalité des séquences codant pour des protéines de 191 génomes de procaryotes (175 génomes de bactéries et 16 génomes d'archaea).

La première étape consiste à calculer les fréquences $\sigma(w_i^p)$ de chaque trinucléotide dans chacune des 3 phases des différents génomes $\mathcal{G}$. La table 4.1. donne un exemple de fréquences obtenues (ici pour le génome *Fusobacterium nucleatum*, AE009951).

Remarque

Les fréquences des 3 codons TAA, TAG et TGA dans la phase 0 sont toujours égales à 0.

Pour chaque groupe G_i de chaque génome $\mathcal{G}$, la fonction $\mathcal{F}$ est appliquée aux 84 ensembles S^i. Les résultats obtenus sont donc les ensembles S^i_{pref} sélectionnés avec leur valeur $\mathcal{F}(\mathrm{S}^i_{pref})$ et le rang associé. La table 4.2. donne pour le génome d'exemple les ensembles S^i_{pref} obtenus, avec leur valeur pour la fonction $\mathcal{F}$ ainsi que leur rang.

En tout, $20 \times 191 = 3820$ groupes G ont été étudiés pour 191 génomes $\mathcal{G}$. Les ensemble préférentiels S_{pref} ont un rang égale à 1, et donc la plus forte valeur parmi les ensembles S pour la fonction $\mathcal{F}$, dans 65% des cas. 80% des ensembles S_{pref} ont un rang ≤ 3. Le génome utilisé en exemple a 15 ensembles S_{pref} avec un rang égal à 1 (table 3.2.). Les 20 ensembles S_{pref} pour un génome $\mathcal{G}$ permettent de construire les 3 ensembles $\mathrm{X}_0(\mathcal{G})$, $\mathrm{X}_1(\mathcal{G})$ et $\mathrm{X}_2(\mathcal{G})$ associés respectivement aux phases 0, 1 et 2 des gènes. Chacun des $3 \times 191 = 573$ ensembles de trinucléotides $\mathrm{X}(\mathcal{G})$ sont examinés afin de déterminer le plus grand code circulaire qu'ils peuvent contenir.

4.4.2 Identification des codes circulaires

L'automate à pétales (2.4.3) correspondant est construit pour chacun des ensembles étudiés afin de tester si il est un code circulaire. Si l'ensemble n'est pas un code circulaire maximale (2.5.), la méthode est répétée sur les sous-ensembles de tailles décroissantes jusqu'à identifier le plus grand code circulaire. Sur les 573 ensembles $\mathcal{X}(\mathcal{G})$ examinés, 446 (78%) sont directement des codes circulaires maximaux. Les 446 ensembles se répartissent

		Fusobacterium nucleatum (AE009951)		
		Phase 0	Phase 1	Phase 2
S_0	AAC	0,71	1,3	**2,47**
	ACA	**2,36**	0,71	1,5
	CAA	1,97	**3,36**	0,71
S_1	AAG	1,58	**5,55**	2,62
	AGA	2,79	1,06	**6,72**
	GAA	**6,99**	2,89	1,36
S_2	AAT	**5,68**	3,25	5,33
	ATA	4,89	**6,1**	1,92
	TAA	0	5,53	**5,68**
S_3	ACC	0,13	0,28	**0,94**
	CCA	**1,21**	0,21	0,26
	CAC	0,17	**0,73**	0,26
S_4	ACG	0,05	**0,12**	0,03
	CGA	0,02	0,08	**0,13**
	GAC	**0,56**	0,62	0,38
S_5	ACT	**2,31**	0,82	1,32
	CTA	0,78	**2,65**	0,58
	TAC	0,5	1,14	**1,74**
S_6	AGC	0,26	0,28	**2,62**
	GCA	**2,65**	0,17	0,3
	CAG	0,19	**2,84**	0,24
S_7	AGG	0,23	0,8	**2,79**
	GGA	**3,76**	0,5	1,09
	GAG	0,89	**2,22**	0,39
S_8	AGT	1,68	0,53	**2,91**
	GTA	**2,25**	1,41	0,35
	TAG	0	**4,44**	1,72
S_9	ATC	0,59	1,17	**1,29**
	TCA	**1,93**	0,52	1,29
	CAT	1,01	**1,2**	0,4
S_{10}	ATG	2,31	**4,76**	0,44
	TGA	0	1,47	**5,34**
	GAT	**4,83**	0,85	0,97
S_{11}	ATT	**4,47**	3,42	4,16
	TTA	5,68	**4,28**	1,87
	TAT	3,93	2,5	**5,31**
S_{12}	CCG	0,02	**0,04**	0,01
	CGC	0,01	0,02	**0,06**
	GCC	**0,27**	0,08	0,2
S_{13}	CCT	**1,26**	0,26	0,15
	CTC	0,07	**0,85**	0,34
	TCC	0,11	0,22	**1,18**
S_{14}	CGG	0	**0,06**	0,07
	GGC	0,21	0,18	**0,5**
	GCG	**0,07**	0,04	0,02
S_{15}	CGT	**0,14**	0,04	0,05
	GTC	0,21	**0,33**	0,25
	TCG	0,06	0,1	**0,11**
S_{16}	CTG	0,11	**2,36**	0,13
	TGC	0,08	0,37	**2,29**
	GCT	**2,47**	0,27	0,33
S_{17}	CTT	1,82	**2,11**	0,97
	TTC	0,64	1,22	**2,16**
	TCT	**1,94**	0,67	0,98
S_{18}	GGT	**1,91**	0,22	0,48
	GTG	0,43	**1,48**	0,1
	TGG	0,62	1,26	**3,01**
S_{19}	GTT	**3,22**	1,2	0,77
	TTG	0,94	**4,66**	0,76
	TGT	0,69	0,69	**2,64**
	AAA	8,54	7,4	8,75
	CCC	0,06	0,07	0,23
	GGG	0,46	0,42	0,47
	TTT	4,28	3,62	5,56

Table 4.1. Fréquences d'occurence des trinucléotides par phase (en %) dans le génome de la bactérie Fusobacterium nucleatum (AE009951). Les ensemble ***S*** contiennent les groupes de trois trinucléotides invariant par permutation. Les fréquences en gras sont les valeurs sélectionnées par la fonction $\mathcal{F}$ (voir Table 3.2).

		Fusobacterium nucleatum (AE009951)	
		Fonction $\mathcal{F}$	Rang Rk
G_0	AAC^0; ACA^1; CAA^2	0,143	
	AAC^1; ACA^2; CAA^0	0,316	
	AAC^2; ACA^0; CAA^1	**0,541**	1
G_1	AAG^0; AGA^1; GAA^2	0,127	
	AAG^1; AGA^2; GAA^0	**0,609**	1
	AAG^2; AGA^0; GAA^1	0,264	
G_2	AAT^0; ATA^1; TAA^2	**0,461**	1
	AAT^1; ATA^2; TAA^0	0,124	
	AAT^2; ATA^0; TAA^1	0,415	
G_3	ACC^0; CCA^1; CAC^2	0,147	
	ACC^1; CCA^2; CAC^0	0,171	
	ACC^2; CCA^0; CAC^1	**0,682**	1
G_4	ACG^0; CGA^1; GAC^2	0,287	
	ACG^1; CGA^2; GAC^0	**0,467**	9
	ACG^2; CGA^0; GAC^1	0,246	
G_5	ACT^0; CTA^1; TAC^2	**0,565**	1
	ACT^1; CTA^2; TAC^0	0,159	
	ACT^2; CTA^0; TAC^1	0,276	
G_6	AGC^0; GCA^1; CAG^2	0,07	
	AGC^1; GCA^2; CAG^0	0,081	
	AGC^2; GCA^0; CAG^1	**0,849**	1
G_7	AGG^0; GGA^1; GAG^2	0,091	
	AGG^1; GGA^2; GAG^0	0,222	
	AGG^2; GGA^0; GAG^1	**0,687**	1
G_8	AGT^0; GTA^1; TAG^2	0,325	
	AGT^1; GTA^2; TAG^0	0,057	
	AGT^2; GTA^0; TAG^1	**0,617**	1
G_9	ATC^0; TCA^1; CAT^2	0,161	
	ATC^1; TCA^2; CAT^0	0,373	
	ATC^2; TCA^0; CAT^1	**0,466**	1
G_{10}	ATG^0; TGA^1; GAT^2	0,224	
	ATG^1; TGA^2; GAT^0	**0,714**	1
	ATG^2; TGA^0; GAT^1	0,062	
G_{11}	ATT^0; TTA^1; TAT^2	**0,398**	7
	ATT^1; TTA^2; TAT^0	0,259	
	ATT^2; TTA^0; TAT^1	0,342	
G_{12}	CCG^0; CGC^1; GCC^2	0,304	
	CCG^1; CGC^2; GCC^0	**0,523**	4
	CCG^2; CGC^0; GCC^1	0,174	
G_{13}	CCT^0; CTC^1; TCC^2	**0,739**	1
	CCT^1; CTC^2; TCC^0	0,162	
	CCT^2; CTC^0; TCC^1	0,099	
G_{14}	CGG^0; GGC^1; GCG^2	0,172	
	CGG^1; GGC^2; GCG^0	**0,479**	8
	CGG^2; GGC^0; GCG^1	0,349	
G_{15}	CGT^0; GTC^1; TCG^2	**0,458**	4
	CGT^1; GTC^2; TCG^0	0,259	
	CGT^2; GTC^0; TCG^1	0,283	
G_{16}	CTG^0; TGC^1; GCT^2	0,095	
	CTG^1; TGC^2; GCT^0	**0,849**	1
	CTG^2; TGC^0; GCT^1	0,056	
G_{17}	CTT^0; TTC^1; TCT^2	0,317	
	CTT^1; TTC^2; TCT^0	**0,5**	1
	CTT^2; TTC^0; TCT^1	0,182	
G_{18}	GGT^0; GTG^1; TGG^2	**0,678**	1
	GGT^1; GTG^2; TGG^0	0,095	
	GGT^2; GTG^0; TGG^1	0,227	
G_{19}	GTT^0; TTG^1; TGT^2	**0,67**	1
	GTT^1; TTG^2; TGT^0	0,172	
	GTT^2; TTG^0; TGT^1	0,159	

Table 4.2. Ensembles preferentiels $\boldsymbol{S}_{\text{pref}}$ associés au génome de la bactérie Fusobacterium nucleatum (AE009951). Les valeurs de la fonction $\mathcal{F}$ pour les 3 ensembles $\boldsymbol{S}_{21}$, $\boldsymbol{S}_{43}$ et $\boldsymbol{S}_{52}$ sont données pour chaque groupe G. Les ensembles sélectionnés $\boldsymbol{S}_{\text{pref}}$ (en gras) sont indiqué avec leur rang parmi les 84 ensembles $\boldsymbol{S}$.

dans les phases potentielles de lecture de la façon suivante : 158 sont associés à la phase 0, i.e. 83% des ensembles $X_0(\mathcal{G})$ sont des codes circulaires, 150 sont associés à la phase 1 (79% des ensembles $X_1(\mathcal{G})$) et 138 sont associés à la phase 2 (72% des ensembles $X_2(\mathcal{G})$. De plus, 110 ensembles $X_0(\mathcal{G})$ sont des codes C^3, c'est à dire que les ensembles $X_0(\mathcal{G})$, $X_1(\mathcal{G})$ et $X_2(\mathcal{G})$, associés à un même génome $\mathcal{G}$ sont tous des codes circulaires. La probabilité qu'un ensemble quelconque de 20 codons non-permutés et non identiques, tel ceux obtenus par la méthode FPTF, soit C^3 est très faible (6.3×10^{-5}). Parmi les ensembles étudiés, 58% (110 sur 191) sont directement C^3.

Pour les $191 - 110 = 81$ génomes de procaryotes dont les codes associés ne sont pas C^3, 49 génomes ont 2 codes circulaires maximaux et 1 non maximal, 18 génomes ont 1 code circulaire maximal et 2 codes non-maximaux et 14 génomes n'ont pas de code circulaire. Pour les $573 - 446 = 127$ ensembles qui ne sont pas des codes circulaires de 20 mots, pratiquement tous (124, i.e. 98%) contiennent un ou plusieurs codes de 19 mots. Les 3 autres ensembles $X(\mathcal{G})$ contiennent des codes circulaires de 18 mots.

Les génomes dont les ensembles $X(\mathcal{G})$ ne sont pas des codes circulaires C^3 ont généralement des ensembles S^i_{pref} dont la valeur $\mathcal{F}(S^i_{pref})$ est très faible, c'est à dire proche de la valeur minimale de $\frac{1}{3}$. La valeur de $\mathcal{F}(S^i_{pref})$ est considérée comme très faible lorsque $\mathcal{F}(S^i_{pref}) - \frac{1}{3} \leq 0.033$ situation rare (2% des 3820 S ensembles analysés). Dans ce cas, le S_{pref} choisi est moins significatif. Ainsi, la sélection d'un autre S_{pref} permet en général d'obtenir un code circulaire maximal. Cette méthode a été appliquée. Des codes de longueur 20 ne différant que de 1 codon (2 pour 3 génomes) par rapport aux premiers codes identifiés ont ainsi pu être obtenus pour les 81 génomes dont les ensembles $X(\mathcal{G})$ n'était pas directement C^3. Ils correspondent aux codes circulaires les plus vraisemblables.

La méthode employée permet donc d'associer un code circulaire C^3 à chacun des 191 génomes de procaryotes. Un certains nombre de codes circulaires obtenus sont communs à plusieurs génomes (table 4.3). En tout 86 nouveaux codes C^3 ont été identifiés. La taille des fenêtres minimales $|w_0|$, $|w_1|$ et $|w_2|$ est évaluée pour chacun des codes circulaires $X_0(\mathcal{G})$, $X_1(\mathcal{G})$ et $X_2(\mathcal{G})$ identifié. La taille des fenêtres des 3×86 codes circulaires varie entre 5 et 13 lettres.

Les tables 4.3 et 4.4 contiennent la liste des codes circulaires identifiés ainsi que les organismes auxquels ils sont associés.

4.4.3 Caractérisation des codes C^3 des procaryotes

La méthode FPTF est basée sur les fréquences de trinucléotides dans chaque phase en considérant la phase préférentielle de chaque trinucléotide et pour chaque phase, le trinucléotide permuté préférentiel. Les gènes de 191 génomes (plus de 500000 gènes pour plus de 540000 kb) ont été analysés. 86 nouveaux codes circulaires ont été identifiés pour les génomes de procaryotes (15 pour les génomes d'archaea et 72 pour les génomes de bactéries, 1 code est commun aux archaea et aux bactéries) (4.4). Les codes C^3 sont spécifiques aux gènes. En effet, ils ne sont pas identifiés dans des séquences génomiques complètes, ni dans des séquences générées aléatoirement.

Les codes des génomes sont associés à un nombre plus ou moins important de génomes. Le code le plus fréquent est associé à 18 génomes et les 11 parmi les 72 codes C^3 sont attribués à la moitié des génomes bactériens. Plusieurs codes n'ont qu'une occurrence unique. La distribution des codes reflètent aussi les choix biologiques de séquençage : les organismes les plus étudiés sont ceux pour lesquels un nombre plus important de séquences est disponible. Les organismes proches du point de vue phylogénétique ont des codes similaires et par conséquent beaucoup de codes n'apparaissant qu'un seul fois sont associés à des génomes d'organismes moins étudiés.

Les codes C^3 associés aux différents chromosomes d'un même organisme ont été comparés. Neuf espèces ont deux chromosomes. Pour six d'entre elles, les codes C^3 obtenus par la méthode PFTF pour chacun des chromosomes sont directement identiques. Les deux chromosomes du *Brucella melitensis* (AE008917 et AE008918) sont associés au code C_8, *Brucella suis* (AE014291 et AE014292) au code C_8, *Deinoccocus radiodurans* (AE000513 et AE001825) au code C_{19}, *Leptospira interrogans* (AE010300, AE010301, AE016823 et AE016824) au code C_9, *Burkholderia pseudomallei* (BX571965 et BX571966) au code C_{31}, et *Vibrio cholerae* (AE003852 et AE003853) au code C_{22}. Pour 3 espèces, les codes C^3 associés à deux chromosomes sont différents : *Vibrio parahaemolyticus* (BA000031 et BA000032) avec les codes C_{58} et C_{15} respectivement, *Vibrio vulnificus* (AE016795 et BA000037, et AE016796 et BA000038) avec les codes C_{15} et C_{23} respectivement, et *Photobacterium profundum* (CR354531 et CR354532) avec les code C_{66} et C_{67}.

De la même manière, plusieurs génomes sont disponibles pour certaines espèces par le séquençage du matériel génétique de plusieurs lignées ou sous espèces différentes. Ainsi, 22 espèces ont plusieurs génomes disponibles et les codes C^3 obtenus sont comparés. Pour 21

Code C^3	Nb de génomes	Nom des génomes (identifiant EMBL, nombre de gènes, taille en kb)
C_0	18	Bordetella bronchisepticaRB50 (BX470250, 5018 g, 5339 kb), Bordetella parapertussis12822 (BX470249, 4627 g, 4774 kb), Bordetella pertussisTohama I (BX470248, 4083 g, 4086 kb), Bradyrhizobium japonicum USDA 110 (BA000040, 8317 g, 9106 kb), Caulobacter crescentus CB15 (AE005673, 3737 g, 4017 kb), Chromobacterium violaceum ATCC 12472 (AE016825, 4407 g, 4751 kb), Desulfovibrio vulgaris subsp vulgarisHildenborough (AE017285, 3380 g, 3571 kb), Leifsonia xyli subsp xyliCTCB07 (AE016822, 2030 g, 2584 kb), Mesorhizobium lotiMAFF303099 (BA000012, 6752 g, 7036 kb), Mycobacterium avium subsp. paratuberculosisk10 (AE016958, 4350 g, 4830 kb), Pseudomonas aeruginosa PAO1 (AE004091, 5566 g, 6264 kb), Ralstonia solanacearum GMI1000 (AL646052, 3442 g, 3716 kb), Rhodopseudomonas palustris CGA009 (BX571963, 4845 g, 5459 kb), Streptomyces avermitilis (BA000030, 7575 g, 9026 kb), Streptomyces coelicolor (AL645882, 7851 g, 8668 kb), Xanthomonas axonopodis pv. citri306 (AE008923, 4312 g, 5176 kb), Xanthomonas campestris pv. campestrisATCC 33913 (AE008922, 4181 g, 5076 kb), Halobacterium sp.NCR-1 (2058 g, 1761 kb)
C_1	14	Erwinia carotovora subsp. atroseptica SCRI1043 (BX950851, 4519 g, 5064 kb), Escherichia coli CFT073 (AE014075, 5380 g, 5231 kb), Escherichia coli K12 MG1655 (U00096, 4255 g, 4640 kb), Escherichia coli O157 H7 EDL933 (AE005174, 5350 g, 5529 kb), Escherichia coli O157:H7 (BA000007, 5362 g, 5498 kb), Nitrosomonas europaea ATCC 19718 (AL954747, 2574 g, 2812 kb), Salmonella enterica CT18 (AL513382, 4606 g, 4809 kb), Salmonella enterica subsp. enterica serovar Typhi Ty2 (AE014613, 4324 g, 4792 kb), Salmonella typhimurium LT2 (AE006468, 4453 g, 4857 kb), Shigella flexneri 2a2457T (AE014073, 4074 g, 4599 kb), Shigella flexneri 2a301 (AE005674, 4434 g, 4607 kb), Thermosynechococcus elongatus BP-1 (BA000039, 2476 g, 2594 kb), Treponema pallidum subsp. pallidumNichols (AE000520, 1031 g, 1138 kb), Wolinella succinogenes DSM 1740 (BX571656, 2044 g, 2110 kb)
C_2	12	Campylobacter jejuni subsp jejuni NCTC 11168 (AL111168, 1654 g, 1641 kb), Chlamydophila caviae GPIC (AE015925, 998 g, 1173 kb), Haemophilus influenzae Rd KW20 (L42023, 1709 g, 1830 kb), Onion yellows phytoplasma OY-M (AP006628, 754 g, 861 kb), Staphylococcus aureus MRSA252 (BX571856, 2834 g, 2903 kb), Staphylococcus aureus MSSA476 (BX571857, 2649 g, 2800 kb), Staphylococcus aureus Mu50 (BA000017, 2699 g, 2879 kb), Staphylococcus aureus MW2 (BA000033, 2632 g, 2820 kb), Staphylococcus aureus N315 (BA000018, 2593 g, 2815 kb), Staphylococcus epidermidis ATCC 12228 (AE015929, 2419 g, 2499 kb), Ureaplasma parvum serovar 3ATCC 700970 (AF222894, 611 g, 752 kb), Yersinia pseudotuberculosis IP 32953 (BX936398, 3983 g, 4745 kb)
C_3	9	Chlamydia muridarum Nigg (AE002160, 904 g, 1073 kb), Chlamydia pneumoniae CWL029 (AE001363, 1052 g, 1230 kb), Chlamydia trachomatis D/UW-3/CX (AE001273, 896 g, 1043 kb), Chlamydophila pneumoniae AR39 (AE002161, 1110 g, 1230 kb), Chlamydophila pneumoniae J138 (BA000008, 1069 g, 1227 kb), Chlamydophila pneumoniae TW-183 (AE009440, 1113 g, 1226 kb), Haemophilus ducreyi 35000HP (AE017143, 1717 g, 1699 kb), Nostoc sp. PCC 7120 (BA000019, 5372 g, 6414 kb), Parachlamydia sp. UWE25 (BX908798, 2031 g, 2414 kb)
C_4	7	Bacillus anthracisAmes (AE016879, 5313 g, 5227 kb), Bacillus anthracisAmes Ancestor (AE017334, 5311 g, 5227 kb), Bacillus anthracisSterne (AE017225, 5288 g, 5229 kb), Bacillus cereus ATCC 10987 (AE017194, 5606 g, 5224 kb), Bacillus cereus ATCC 14579 (AE016877, 5234 g, 5412 kb), Bacillus cereus ZK (CP000001, 5134 g, 5301 kb), Bacillus thuringiensis serovar konkukian97-27 (AE017355, 5117 g, 5238 kb)
C_5	6	Lactobacillus johnsonii NCC 533 (AE017198, 1821 g, 1993 kb), Mycoplasma mycoides subsp. mycoides SC PG1 (BX293980, 1016 g, 1212 kb), Mycoplasma pulmonisUAB CTIP (AL445566, 782 g, 964 kb), Rickettsia prowazekiiMadrid E (AJ235269, 835 g, 1112 kb), Rickettsia typhiWilmington (AE017197, 841 g, 1111 kb), Wolbachia endosymbiont of Drosophila melanogaster (AE017196, 1195 g, 1268 kb)
C_6	4	Mycobacterium bovis AF2122/97 (BX248333, 3953 g, 4345 kb), Mycobacterium tuberculosis CDC1551 (AE000516, 4187 g, 4404 kb), Mycobacterium tuberculosis H37Rv (AL123456, 3999 g, 4412 kb), Pseudomonas putida KT2440 (AE015451, 5350 g, 6182 kb)
C_7	4	Bartonella quintanaToulouse (BX897700, 1308 g, 1581 kb), Lactococcus lactis subsp. lactis IL1403 (AE005176, 2266 g, 2366 kb), Streptococcus agalactiae 2603V/R (AE009948, 2124 g, 2160 kb), Streptococcus agalactiae NEM316 (AL732656, 2134 g, 2211 kb)
C_8	4	Brucella melitensis 16M chromosome I (AE008917, 2059 g, 2117 kb), Brucella melitensis 16M chromosome II (AE008918, 1139 g, 1178 kb), Brucella suis 1330 chromosome I (AE014291, 2124 g, 2108 kb), Brucella suis 1330 chromosome II (AE014292, 1148 g, 1207 kb)
C_9	4	Leptospira interrogans serovar CopenhageniFiocruz L1-130 chromosome I (AE016823, 3394 g, 4277 kb), Leptospira interrogans serovar CopenhageniFiocruz L1-130 chromosome II (AE016824, 264 g, 350 kb), Leptospira interrogans serovar lai56601 chromosome I (AE010300, 4358 g, 4332 kb), Leptospira interrogans serovar lai56601 chromosome II (AE010301, 367 g, 359 kb)
C_{10}	4	Streptococcus pyogenes M1 GAS (AE004092, 1696 g, 1852 kb), Streptococcus pyogenes MGAS315 (AE014074, 1865 g, 1901 kb), Streptococcus pyogenes MGAS8232 (AE009949, 1845 g, 1895 kb), Streptococcus pyogenes SSI-1 (BA000034, 1861 g, 1894 kb)
C_{11}	3	Agrobacterium TumefaciensC58 circular Washington (AE008688, 2785 g, 2841 kb), Agrobacterium tumefaciensC58 linear chromosome (AE008689, 1876 g, 2076 kb), Sinorhizobium meliloti 1021 (AL591688, 3341 g, 3654 kb)
C_{12}	3	Borrelia burgdorferi B31 (AE000783, 850 g, 911 kb), Borrelia garinii PBi (CP000013, 832 g, 904 kb), Prochlorococcus marinus CCMP1986 (BX548174, 1717 g, 1658 kb)
C_{13}	3	Neisseria meningitidis MC58 (AE002098, 2025 g, 2272 kb), Neisseria meningitidisZ2491 (AL157959, 2121 g, 2184 kb), Pirellula sp.1 (BX119912, 7325 g, 7146 kb)
C_{14}	3	Photorhabdus luminescens subsp. laumondii TTO1 (BX470251, 4905 g, 5689 kb), Streptococcus pneumoniae R6 (AE007317, 2043 g, 2039 kb), Streptococcus pneumoniae TIGR4 (AE005672, 2094 g, 2161 kb)
C_{15}	3	Vibrio parahaemolyticus RIMD 2210633 chromosome 2 (BA000032, 1752 g, 1877 kb), Vibrio vulnificus CMCP6 chromosome I (AE016795, 2972 g, 3282 kb), Vibrio vulnificus YJ016 chromosome I (BA000037, 3262 g, 3355 kb)
C_{16}	3	Yersinia pestis biovar Medievalis91001 (AE017042, 3895 g, 4595 kb), Yersinia pestis CO92 (AL590842, 4034 g, 4654 kb), Yersinia pestis KIM (AE009952, 4090 g, 4601 kb)
C_{17}	3	Mesoplasma florum L1 (AE017263, 683 g, 793 kb), Mycoplasma mobile 163K (AE017308, 633 g, 777 kb), Mycoplasma penetrans HF-2 (BA000026, 1037 g, 1359 kb)
C_{18}	3	Buchnera aphidicolaSg (AE013218, 545 g, 641 kb), Buchnera aphidicola APS (BA000003, 564 g, 641 kb), Buchnera aphidicolaBp (AE016826, 504 g, 616 kb)
C_{19}	2	Deinococcus radiodurans R1 chromosome 1 (AE000513, 2579 g, 2649 kb), Deinococcus radiodurans R1 chromosome 2 (AE001825, 357 g, 412 kb)
C_{20}	2	Helicobacter pylori 26695 (AE000511, 1566 g, 1668 kb), Helicobacter pylori J99 (AE001439, 1505 g, 1644 kb)
C_{21}	2	Xylella fastidiosa 9a5c (AE003849, 2767 g, 2679 kb), Xylella fastidiosa Temecula1 (AE009442, 2034 g, 2520 kb)
C_{22}	2	Vibrio cholerae O1 biovar N16961 chromosome I (AE003852, 2736 g, 2961 kb), Vibrio cholerae O1 biovar N16961 chromosome II (AE003853, 1092 g, 1072 kb)
C_{23}	2	Vibrio vulnificus CMCP6 chromosome II (AE016796, 1565 g, 1845 kb), Vibrio vulnificus YJ016 chromosome II (BA000038, 1697 g, 1857 kb)
C_{24}	2	Chlorobium tepidum TLS (AE006470, 2252 g, 2155 kb), Geobacter sulfurreducens PCA (AE017180, 3447 g, 3814 kb)
C_{25}	2	Enterococcus faecalis V583 (AE016830, 3113 g, 3218 kb), Rickettsia conoriiMalish 7 (AE006914, 1375 g, 1269 kb)
C_{26}	2	Helicobacter hepaticus ATCC 51449 (AE017125, 1875 g, 1799 kb), Streptococcus mutans UA159 (AE014133, 1960 g, 2031 kb)

Code C^3	Nb de génomes	Nom des génomes (identifiant EMBL, nombre de gènes, taille en kb)
C_{27}	2	Tropheryma whipplei TW08/27 (BX072543, 788 g, 926 kb), Tropheryma whippleiTwist (AE014184, 808 g, 927 kb)
C_{28}	2	Gloeobacter violaceus PCC 7421 (BA000045, 4430 g, 4659 kb), Pseudomonas syringae pv. tomatoDC3000 (AE016853, 5471 g, 6397 kb)
C_{29}	2	Listeria monocytogenes EGD-e (AL591824, 2855 g, 2945 kb), Listeria monocytogenes4b F2365 (AE017262, 2822 g, 2905 kb)
C_{30}	2	Corynebacterium glutamicum ATCC 13032 (BA000036, 3099 g, 3309 kb), Corynebacterium glutamicum ATCC 13032 4-5 (BX927147, 3058 g, 3283 kb)
C_{31}	2	Burkholderia pseudomallei K96243 chr. 1 (BX571965, 3503 g, 4075 kb), Burkholderia pseudomallei K96243 chr. 2 (BX571966, 2445 g, 3173 kb)
C_{32}	1	Thermotoga maritima MSB8 (AE000512, 1846 g, 1861 kb)
C_{33}	1	Aquifex aeolicus VF5 (AE000657, 1522 g, 1551 kb)
C_{34}	1	Clostridium acetobutylicum ATCC 824 (AE001437, 3672 g, 3941 kb)
C_{35}	1	Pasteurella multocida PM70 (AE004439, 2014 g, 2257 kb)
C_{36}	1	Thermoanaerobacter tengcongensis MB4 (AE008691, 2588 g, 2689 kb)
C_{37}	1	Fusobacterium nucleatum subsp. nucleatum ATCC 25586 (AE009951, 2068 g, 2174 kb)
C_{38}	1	Bifidobacterium longum NCC2705 (AE014295, 1727 g, 2257 kb)
C_{39}	1	Shewanella oneidensis MR-1 (AE014299, 4630 g, 4970 kb)
C_{40}	1	Mycoplasma gallisepticum R (AE015450, 726 g, 996 kb)
C_{41}	1	Porphyromonas gingivalis W83 (AE015924, 1909 g, 2343 kb)
C_{42}	1	Clostridium tetani E88 (AE015927, 2373 g, 2799 kb)
C_{43}	1	Bacteroides thetaiotaomicron VPI-5482 (AE015928, 4778 g, 6260 kb)
C_{44}	1	Coxiella burnetii RSA 493 (AE016828, 2010 g, 1995 kb)
C_{45}	1	Prochlorococcus marinus subsp. marinusCCMP1375 (AE017126, 1882 g, 1751 kb)
C_{46}	1	Thermus thermophilus HB27 (AE017221, 1982 g, 1895 kb)
C_{47}	1	Treponema denticola ATCC 35405 (AE017226, 2767 g, 2843 kb)
C_{48}	1	Propionibacterium acnes KPA171202 (AE017283, 2297 g, 2560 kb)
C_{49}	1	Bacillus subtilis168 (AL009126, 4109 g, 4215 kb)
C_{50}	1	Mycobacterium leprae TN (AL450380, 2720 g, 3268 kb)
C_{51}	1	Listeria innocua Clip11262 (AL592022, 2981 g, 3011 kb)
C_{52}	1	Lactobacillus plantarum WCFS1 (AL935263, 3051 g, 3308 kb)
C_{53}	1	Bacillus halodurans C-125 (BA000004, 4066 g, 4202 kb)
C_{54}	1	Clostridium perfringens13 (BA000016, 2660 g, 3031 kb)
C_{55}	1	Wigglesworthia glossinidia endosymbiont of Glossina brevipalpis (BA000021, 615 g, 698 kb)
C_{56}	1	Synechocystis sp. PCC 6803 (BA000022, 3171 g, 3573 kb)
C_{57}	1	Oceanobacillus iheyensis (BA000028, 3496 g, 3631 kb)
C_{58}	1	Vibrio parahaemolyticus RIMD 2210633 chromosome 1 (BA000031, 3080 g, 3289 kb)
C_{59}	1	Corynebacterium efficiens YS-314 (BA000035, 2942 g, 3147 kb)
C_{60}	1	Corynebacterium diphtheriae NCTC 13129 (BX248353, 2400 g, 2489 kb)
C_{61}	1	Blochmannia floridanus (BX248583, 589 g, 706 kb)
C_{62}	1	Synechococcus sp. WH8102 (BX548020, 2527 g, 2434 kb)
C_{63}	1	Prochlorococcus marinus MIT 9313 (BX548175, 2274 g, 2411 kb)
C_{64}	1	Bdellovibrio bacteriovorus HD100 (BX842601, 3583 g, 3783 kb)
C_{65}	1	Bartonella henselaeHouston-1 (BX897699, 1612 g, 1931 kb)
C_{66}	1	Photobacterium profundum SS9 chromosome 1 (CR354531, 3416 g, 4085 kb)
C_{67}	1	Photobacterium profundum SS9 chromosome 2 (CR354532, 1997 g, 2238 kb)
C_{68}	1	Desulfotalea psychrophila LSv54 (CR522870, 3118 g, 3523 kb)
C_{69}	1	Acinetobacter sp. ADP1 (CR543861, 3325 g, 3599 kb)
C_{70}	1	Mycoplasma genitalium G-37 (L43967, 480 g, 580 kb)
C_{71}	1	Mycoplasma pneumoniae M129 (U00089, 688 g, 816 kb)
C_{72}	2	Archeoglobus fulgidus (2407 g, 1989 kb), Aeropyrum pernix (2694 g, 1916 kb)
C_{73}	1	Halobacterium sp.NCR-1 (2058 g, 1761 kb)
C_{74}	1	Methanoccocus jannashii (1709 g, 1444 kb)
C_{75}	1	Methanopyrus kandleri (1678 g, 1492 kb)
C_{76}	1	Methanosarcina acetivorans (4440 g, 4162 kb)
C_{77}	1	Methanosarcina mazei (3371 g, 3065 kb)
C_{78}	1	Methanothermobacter thermautotrophicus (1868 g, 1575 kb)
C_{79}	1	Pyrobaculum aerophilum (2605 g, 1968 kb)
C_{80}	1	Pyroccocus abyssi (1762 g, 1606 kb)
C_{81}	1	Pyroccocus furiosus (2060 g, 1740 kb)
C_{82}	1	Pyroccocus horikoshii (2058 g, 1704 kb)
C_{83}	1	Sulfolobus solfataricus (2994 g, 2525 kb)
C_{84}	1	Sulfolobus tokodaii (2826 g, 2276 kb)
C_{85}	1	Thermoplasma acidophilium (1478 g, 1359 kb)

Table 4.3. Liste des 191 genomes de bactéries $_G$ associés à 86 C^3 codes.

Codes C^3	Nb de génomes	Liste des 20 trinucléotides
C_0	17	AAC AAG AAT ACC GAC TAC CAG GAG TAG ATC ATG TAT GCC CTC GGC GTC CTG TTC GTG TTG
C_1	14	AAC GAA AAT ACC GAC TAC CAG GAG GTA ATC GAT ATT GCC CTC GCG GTC CTG CTT GTG GTT
C_2	12	CAA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GGC CGT GCT TCT GGT GTT
C_3	9	CAA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GCG CGT GCT TCT GGT GTT
C_4	7	ACA GAA AAT CCA ACG ACT GCA GGA GTA CAT GAT ATT CCG CCT GCG CGT GCT TCT GGT GTT
C_5	6	ACA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GGC CGT GCT TCT GGT GTT
C_6	4	AAC AAG AAT ACC GAC TAC CAG GAG TAG ATC ATG TAT GCC CTC GGC GTC CTG TTC GTG GTT
C_7	4	AAC AAG AAT ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC CTC GGC GTC CTG TTC GTG TTG
C_8	4	ACA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GCG CGT GCT CTT GGT GTT
C_9	4	AAC AAG AAT ACC GAC TAC CAG GAG GTA ATC GAT TAT GCC CTC GGC GTC CTG CTT GTG GTT
C_{10}	4	ACA GAA AAT ACC GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GCG GTC GCT TCT GGT GTT
C_{11}	3	ACA GAA AAT ACC GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GCG CGT GCT CTT GGT GTT
C_{12}	3	ACA GAA AAT CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GGC GTC GCT TCT GGT GTT
C_{13}	3	AAC GAA AAT ACC GAC TAC AGC GAG GTA ATC ATG ATT GCC CTC GGC GTC CTG TTC GTG TTG
C_{14}	3	CAA GAA AAT CAC GAC ACT GCA GAG GTA CAT GAT ATT GCC CCT GCG CGT GCT CTT GGT GTT
C_{15}	3	AAC GAA AAT ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CCT GCG GTC GCT CTT GGT GTT
C_{16}	3	CAA GAA AAT CAC GAC CTA CAG GAG GTA ATC GAT ATT GCC CTC GCG GTC CTG TTC GTG GTT
C_{17}	3	ACA GAA AAT CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GGC CGT GCT TCT GGT GTT
C_{18}	3	CAA GAA AAT CCA CGA ACT GCA GGA GTA CAT GAT ATT GCC CCT GGC CGT GCT TCT GGT GTT
C_{19}	2	AAC AAG AAT ACC GAC TAC CAG GAG TAG ATC ATG ATT GCC CTC GGC GTC CTG TTC GTG TTG
C_{20}	2	AAC GAA AAT ACC GAC ACT AGC GAG GTA ATC GAT ATT GCC TCC GGC GTC GCT TCT GTG TTG
C_{21}	2	AAC GAA AAT ACC GAC CTA CAG GAG GTA ATC GAT ATT GCC CTC GGC GTC CTG TTC GTG GTT
C_{22}	2	AAC GAA AAT ACC GAC CTA CAG GAG GTA ATC GAT ATT GCC CTC GCG GTC CTG TTC GTG GTT
C_{23}	2	AAC GAA AAT ACC GAC TAC CAG GAG GTA ATC GAT ATT GCC CTC GCG GTC CTG TTC GTG GTT
C_{24}	2	AAC AAG AAT ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC CTC GGC GTC CTG TTC GTG GTT
C_{25}	2	ACA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GGC GTC GCT TCT GGT GTT
C_{26}	2	CAA GAA AAT CCA GAC ACT GCA GAG GTA CAT GAT ATT GCC CCT GGC CGT GCT CTT GGT GTT
C_{27}	2	ACA GAA ATA ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CTC GCG GTC GCT CTT GGT GTT
C_{28}	2	AAC GAA AAT ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC CTC GGC GTC CTG TTC GTG TTG
C_{29}	2	ACA GAA AAT CCA GAC ACT GCA GGA GTA ATC GAT ATT GCC CCT GGC GTC GCT CTT GGT GTT
C_{30}	2	AAC GAA AAT ACC GAC TAC GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT TTC GTG GTT
C_{31}	2	AAC AAG AAT CAC GAC TAC CAG GAG TAG ATC ATG TAT GCC CTC GGC GTC CTG TTC GTG TTG
C_{32}	1	AAC GAA ATA ACC GAC TAC GCA GAG GTA ATC ATG TTA GCC CTC GCG GTC CTG TTC GTG GTT
C_{33}	1	AAC GAA ATA CAC GAC TAC GCA GAG GTA ATC ATG ATT GCC CTC GCG GTC GCT TTC GTG GTT
C_{34}	1	ACA GAA AAT CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GCG GTC GCT CTT GGT GTT
C_{35}	1	CAA GAA AAT CCA ACG ACT GCA GAG GTA CAT GAT ATT GCC CCT GCG CGT GCT CTT GGT GTT
C_{36}	1	ACA GAA ATA ACC GAC ACT GCA GGA GTA ATC GAT ATT GCC CCT GCG GTC GCT TCT GTG GTT
C_{37}	1	ACA GAA AAT CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GCG CGT GCT TCT GGT GTT
C_{38}	1	AAC AAG AAT ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC TCC GGC GTC CTG TTC GTG TTG
C_{39}	1	ACA GAA AAT ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC TCC GCG GTC GCT TTC GGT GTT
C_{40}	1	CAA GAA AAT CAC GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GGC GTC GCT CTT GGT GTT
C_{41}	1	AAC GAA AAT ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC CTC GGC GTC CTG TTC GTG GTT
C_{42}	1	ACA GAA ATA CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GCG CGT GCT TCT GGT GTT
C_{43}	1	AAC GAA AAT ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CCT GGC GTC GCT CTT GTG GTT
C_{44}	1	CAA GAA AAT CAC GAC ACT GCA GAG GTA CAT GAT ATT GCC CCT GCG GTC GCT CTT GTG GTT
C_{45}	1	ACA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GGC CGT GCT CTT GGT GTT
C_{46}	1	AAC AAG ATA CAC GAC TAC CAG GAG TAG ATC ATG ATT GCC CTC GCG GTC CTG TTC GTG TTG
C_{47}	1	ACA GAA ATA ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CCT GGC GTC GCT CTT GTG GTT
C_{48}	1	AAC AAG AAT ACC GAC TAC CAG GAG GTA ATC GAT ATT GCC CTC GGC GTC CTG TTC GTG GTT
C_{49}	1	AAC GAA AAT ACC GAC CTA GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC CTG CTT GTG GTT
C_{50}	1	AAC GAA AAT ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC CTC GCG GTC CTG TTC GTG TTG
C_{51}	1	ACA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT GCC CCT GGC GTC GCT CTT GGT GTT
C_{52}	1	ACA GAA AAT ACC ACG ACT GCA GAG GTA ATC GAT ATT GCC TCC GCG GTC GCT TTC GGT GTT
C_{53}	1	CAA GAA ATA CCA ACG CTA GCA GAG GTA ATC GAT ATT GCC CTC GCG GTC GCT CTT GTG GTT
C_{54}	1	ACA GAA ATA CCA GAC ACT GCA GGA GTA TCA GAT TTA GCC CCT GGC GTC GCT TCT GGT GTT
C_{55}	1	ACA GAA ATA CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GGC CGT GCT TCT GGT GTT
C_{56}	1	AAC GAA AAT ACC GAC ACT CAG GAG GTA ATC GAT ATT GCC CTC GCG GTC GCT CTT GTG GTT
C_{57}	1	CAA GAA AAT CCA ACG ACT GCA GGA GTA CAT GAT ATT GCC CCT GCG CGT GCT TCT GGT GTT
C_{58}	1	AAC GAA AAT CAC GAC CTA GCA GAG GTA CAT GAT ATT GCC CCT GCG GTC GCT CTT GGT GTT
C_{59}	1	AAC AAG AAT ACC GAC TAC CAG GAG TAG ATC GAT TAT GCC CTC GCG GTC CTG TTC GTG GTT
C_{60}	1	AAC GAA AAT ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT CTT GTG GTT
C_{61}	1	CAA GAA AAT CCA CGA ACT GCA GGA GTA CAT GAT ATT CCG CCT GCG CGT GCT TCT GGT GTT
C_{62}	1	AAC AAG AAT ACC GAC TAC CAG GAG GTA ATC GAT TAT GCC CTC GGC GTC CTG TTC GTG GTT
C_{63}	1	AAC GAA AAT ACC GAC CTA GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT CTT GTG GTT
C_{64}	1	AAC GAA AAT ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC TCC GCG GTC CTG TTC GTG GTT
C_{65}	1	ACA GAA AAT CCA CGA ACT GCA GGA GTA CAT GAT ATT GCC CCT GCG CGT GCT CTT GGT GTT
C_{66}	1	CAA GAA AAT CCA GAC ACT GCA GAG GTA CAT GAT ATT GCC CCT GCG CGT GCT CTT GGT GTT
C_{67}	1	ACA GAA AAT ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CCT GCG GTC GCT CTT GGT GTT
C_{68}	1	AAC GAA AAT ACC GAC TAC GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT CTT GGT GTT
C_{69}	1	CAA GAA AAT CCA GAC CTA GCA GAG GTA CAT GAT ATT GCC CCT GCG CGT GCT CTT GGT GTT
C_{70}	1	CAA GAA AAT CCA GAC ACT GCA GGA GTA CAT GAT ATT CGC CCT GGC CGT GCT CTT GGT GTT
C_{71}	1	AAC GAA AAT CAC GAC ACT CAG GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT TTC GGT GTT

Codes C^3	Nb de génomes	Liste des 20 trinucléotides
C_{72}	2	AAC AAG ATA ACC GAC TAC AGC GAG GTA ATC ATG ATT GCC CTC GCG GTC CTG TTC GTG GTT
C_{73}	1	ACA GAA ATA CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GCG GTC GCT TCT GGT GTT
C_{74}	1	AAC AAG ATA ACC GAC TAC CAG GAG GTA ATC ATG ATT GCC CTC GCG GTC CTG TTC GTG TTG
C_{75}	1	AAC GAA ATA ACC GAC TAC GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT CTT GTG GTT
C_{76}	1	AAC GAA ATA ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT CTT GTG GTT
C_{77}	1	AAC AAG ATA ACC GAC TAC GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT TTC GTG GTT
C_{78}	1	AAC GAA ATA ACC GAC TAC GCA GAG GTA ATC ATG ATT GCC CTC GCG GTC CTG TTC GTG GTT
C_{79}	1	AAC AAG ATA ACC GAC TAC GCA GAG GTA ATC GAT ATT GCC CTC GCG GTC GCT TTC GTG GTT
C_{80}	1	ACA GAA ATA CCA GAC CTA GCA GAG GTA ATC GAT ATT GCC CTC GCG GTC GCT CTT GTG GTT
C_{81}	1	ACA GAA ATA CCA GAC CTA GCA GAG GTA ATC GAT ATT GCC CTC GCG GTC GCT TTC GTG GTT
C_{82}	1	ACA GAA ATA CCA GAC ACT GCA GAG GTA TCA GAT ATT GCC CCT GCG GTC GCT TCT GGT GTT
C_{83}	1	ACA GAA ATA CCA GAC ACT GCA GGA GTA TCA GAT ATT GCC CCT GGC GTC GCT TCT GGT GTT
C_{84}	1	AAC AAG ATA ACC GAC TAC AGC GAG GTA ATC ATG ATT GCC CTC GGC GTC CTG TTC GTG GTT
C_{85}	1	AAC GAA ATA ACC GAC ACT GCA GAG GTA ATC GAT ATT GCC CTC GGC GTC GCT CTT GGT GTT

Table 4.4. Liste des 86 codes C^3 dans les 191 génomes de bactéries $\mathcal{G}$ pour chaque code C^3 .

espèces, les codes circulaires identifiés sont identiques. Seul une espèce, le *Prochlorococcus marinus*, a différents codes (C^{12}, C^{45} et C^{63}) associés à ces différentes lignées (AE017126, BX548174, BX548175).

Plusieurs codes C^3 de bactéries sont proches du code complémentaire $\mathcal{X}$, identifié à partir de grandes populations de gènes d'eucaryotes et de procaryotes. De plus, le code $\mathcal{X}(PRO)$, obtenu par l'application de la méthode FPTF à la totalité des gènes des 191 génomes de procaryotes ne diffère que d'un trinucléotide de celui de $\mathcal{X}$: GGT de $\mathcal{X}(PRO)$ est remplacé par GTG dans $\mathcal{X}$.

Plusieurs codes C^3 peuvent avoir dérivé par mutations à partir du code $\mathcal{C}^3$ X qui semble être le code circulaire le plus général (présence dans les séquences de procaryotes et d'eucaryotes) et qui reste le seul code identifié à être auto-complémentaire. Un modèle d'évolution est présenté dans le chapitre 5 qui permet d'examiner des transitions possibles au cours de l'évolution entre le code commun $\mathcal{X}$ et les codes C^3 associés aux génomes d'archaea (tous les génomes d'archaea sont associés aux codes de $\mathcal{C}^{73}$ à $\mathcal{C}^{81}$, un archaea étant associé à $\mathcal{C}_0$).

4.4.4 Trinucléotides des codes C^3

Les occurrences des trinucléotides dans les 86 codes C^3 ont été évaluées. Les codons rares et fréquents dans les différents codes $X_0(\mathcal{G})$ sont les suivants :

- 10 codons sont absents, le nombre Nb de trinucléotide dans les 86 codes est égal à 0 : AGA, AGG, AGT, CGG, TAA, TCG, TGA, TGC, TGG, TGT
- 18 codons sont très rares, $0 < Nb \leq 21$ (premier quart) : AAG, ACG, AGC ATA, ATG, CAA, CAC, CCG, CGA, CGC, CTA, TAG, TAT, TCA, TCC, TCT, TTA, TTG
- 18 sont rares, $22 \leq Nb \leq 43$ (second quart) : AAC, ACA, ACC, CAG, CAT, CCA, CCT, CGT, CTC, CTG, CTT, GCG, GGA, GGC, GGT, GTG, TAC, TTC
- 6 sont courants, $44 \leq Nb \leq 64$ (troisième quart) : ACT, ATC, GAG, GCA, GCT, GTC
- 8 sont très courants, $Nb > 64$ (troisième quart) : AAT, ATT, GAA, GAC, GAT, GCC, GTA, GTT.

Les occurrences des 4 types de nucléotides à chaque position des trinucléotides des 86 codes $\mathcal{X}(\mathcal{G})$ des procaryotes sont les suivantes :

- dans le premier site des trinucléotides, tous les nucléotides apparaissent à l'exception de C dans 3 codes et de T dans 28 codes.

- dans le second site des trinucléotides, tous les nucléotides apparaissent à l'exception de G dans 18 codes.
- dans le troisième site des trinucléotides, tous les nucléotides apparaissent à l'exception de A dans 5 codes, C dans 2 codes et de G dans 14 codes.

Les 4 codes C_{44}, C_{53}, C_{56} et C_{81} n'ont pas de lettre T à la 1^{ere} position des trinucléotides ni de lettre G à la 2^{de} position. Pour 5 codes, C_{29}, C_{40}, C_{45}, C_{51} et C_{70}, il n'y a ni lettre T à la 1^{ere} position ni la lettre G à la 3^{eme} position. Dans le code, C_{59}, il n'y a ni lettre G à la 2^{de} position ni lettre A à la 3^{eme} position.

La distribution des trinucléotides et les règles sur les apparitions de nucléotides dans les trinucléotides pour les 86 codes $X_1(\mathcal{G})$ et $X_2(\mathcal{G})$ peuvent être déduit par permutation de celles de $X_0(\mathcal{G})$.

4.4.5 Les codes C^3 sur les alphabets réduits

Considérons les 8 trinucléotides sur l'alphabet R/Y. Chacun de ces trinucléotides peut être instancié en 8 trinucléotides sur l'alphabet {A,C,G,T}. Les 3 codes circulaires $X_0(\mathcal{G})$, $X_1(\mathcal{G})$ et $X_2(\mathcal{G})$ contiennent respectivement les 20 trinucléotides associés préférentiellement aux phases 0, 1 et 2. Par conséquent, une phase préférentielle pour les 8 trinucléotides sur l'alphabet réduit R/Y peut être déduit en associant à chaque trinucléotide sur R/Y la phase moyenne des 8 trinucléotides instanciés.

Par exemple, pour le code C_0, RRY correspond aux 8 trinucléotides AAC, AAT, AGC, AGT, GAC, GAT, GGC, GGT qui sont associés respectivement aux phases 0, 0, 1, 1, 0, 2, 0, 2. Le codon RRY dans le code C_0 est donc associé à la phase 0, celle-ci étant majoritaire.

Tous les 72 codes de bactéries ont le trinucléotide RYY associé à la phase 0. Cette propriété existe également pour le code commun $\mathcal{X}$ et pour le code des mitochondries [AM97]. Les trinucléotides permutés YYR et YRY sont logiquement associés à la phase 1 et 2 respectivement. Pour 45 codes de bactéries, le trinucléotide RRY est associé préférentiellement à la phase 0, pour 16 codes, c'est le trinucléotide RYR qui est associé à la phase 0. Pour les 11 autres codes, RRY et RYR apparaissent en quantité égale en phase 0. Il n'y a pas de phase préférentielle associé aux trinucléotides RRR et YYY. Par conséquent, les codes de bactérie sont conformes au motif RNY={RRY,RYY} avec N = {R,Y} [ES78].

Les trinucléotides sur l'alphabet réduit R/Y des codes des archaea ne sont pas conforme au motif RNY. En effet, RYY et RYR sont associés préférentiellement à la phase 0. Par conséquent, le motif associé aux codes des archaea est RYN={RYY, RYR} qui est donc le permuté du motif classique RNY. La recherche de phases préférentielles associées aux codes C^3 sur les autres alphabets réduit K/M (K = ceto = {G, T} et M = amino = {A, C}) et S/W (S = strong interaction = {C,G} et W = weak interation = {A, T}) se révèle peu concluante, aussi bien pour les archaea que pour les bactéries.

En conclusion, l'alphabet R/Y comporte plus d'information génétique pour retrouver la phase de lecture que les alphabets K/N et S/W.

4.4.6 Acides aminés codés par les $\mathcal{C}^3$ des procaryotes

Les ensemble $X_0(\mathcal{G})$ sont associés à la phase 0. Les 20 trinucléotides de ces ensembles codent pour des acides aminés.

Deux acides aminés ne sont jamais codés par les codons des 86 codes de procaryotes : la cystéine et le tryptophane. Ces deux acides aminés ont une structure chimique complexe en terme de nombre d'atomes par cycles. La cystéine à un rôle particulier. Elle peut former des ponts disulfure par réaction avec d'autres cystéines. Le tryptophane est le seul acide aminé avec deux cycles de carbones. De plus, le tryptophane est représenté par un seul trinucléotide ce qui réduit sa probabilité d'apparition dans un code.

Cinq acides aminés sont présents dans tous les codes de procaryotes : l'alanine, l'acide aspartique, l'acide glutamique, l'isoleucine et la valine. La thréonine est fortement présente, car absente uniquement dans les codes C_{16}, C_{29}, C_{31}, C_{16}, C_{46} et C_{69}. L'alanine et l'acide aspartique représentent le groupe complet des acides aminés négativement chargés. Ces 6 acides aminés sont représentés de manière égale dans les deux classes d'aminoacyl-tRNA synthétases, la classe I contenant l'acide glutamique, l'isoleucine et la valine, et la classe II l'alanine, l'acide aspartique et la thréonine. Les ensembles C^3 des bactéries codent entre 8 (pour C_{27} et C_{81}) et 15 acides aminés (C_{36}).

4.5 Méthode de factorisation pour retrouver la phase de lecture des gènes de procaryotes à partir des ensembles de nucléotides X identifiés

A partir d'une position quelconque, en parcourant quelques nucléotides, il est possible de retrouver la phase de lecture de séquences composées uniquement de mots d'un code circulaire $\mathrm{X}(\mathcal{G})$. Cependant, les gènes réels ne sont pas uniquement composés à partir de la concaténation de 20 mots. Tous les trinucléotides à l'exception des codon stop peuvent apparaître en phase 0. Par conséquent les codes circulaires ne peuvent servir à retrouver la phase de lecture dans le cas général. Cependant, les codes C^3 identifiés contiennent des informations sur les fréquences d'occurrences des trinucléotides dans les 3 phases potentielles des gènes de procaryotes et possèdent une structure particulière. Ainsi, une méthode simple de factorisation a été développée pour estimer la probabilité de retrouver la phase de lecture à partir de la lecture de très courtes suites de nucléotide et de l'information résumée par les ensembles C^3.

Afin d'obtenir des résultats stables et statistiquement significatifs, la méthode a été testée sur un grand nombre de mots de longueurs différentes extraits à partir de positions quelconques de séquences génétiques choisies aléatoirement. Soit w_0 un mot extrait d'une séquence nucléique d'un génome $\mathcal{G}$. Comme w_0 est extrait à partir d'une position quelconque, sa phase réelle, i.e. son décalage par rapport au début de la séquence modulo 3, est inconnue. Le code C^3 $\mathrm{X}_0(\mathcal{G})$ est l'ensemble des trinucléotides associés préférentiellement à la phase 0 qui est obtenu par l'application de la méthode FPTF au génome $\mathcal{G}$. Par permutation circulaire de chaque mot de l'ensemble $\mathrm{X}_0(\mathcal{G})$, il est possible de déduire les ensembles $\mathrm{X}_1(\mathcal{G})$ et $\mathrm{X}_2(\mathcal{G})$ associés respectivement aux phases 1 et 2. La phase du mot w_0 n'étant pas connue, w_0 sera comparé à ses deux mots permutés w_1 et w_2 déduits de w_0 auxquels on enlève respectivement 1 et 2 nucléotides. La fin des mots w_0, w_1 et w_2 est tronquée de telle sorte que leur longueur soit égale à 0 modulo 3. Les mots w_0, w_1 et w_2 sont factorisés en mots des codes $\mathrm{X}_0(\mathcal{G})$, $\mathrm{X}_1(\mathcal{G})$ et $\mathrm{X}_2(\mathcal{G})$. Soit $\mathcal{N}(w, \mathrm{X}(\mathcal{G}))$ le nombre de mots de $\mathcal{X}(\mathcal{G})$ dans la phase 0 du mot w. Les valeurs $\mathcal{V}_0(\mathcal{G})$, $\mathcal{V}_1(\mathcal{G})$ et $\mathcal{V}_2(\mathcal{G})$ sont définies tel que

$$\mathcal{V}_0(\mathcal{G}) = \mathcal{N}(w_0, \mathrm{X}_0(\mathcal{G})) + \mathcal{N}(w_1, \mathrm{X}_1(\mathcal{G})) + \mathcal{N}(w_2, \mathrm{X}_2(\mathcal{G}))$$

$$\mathcal{V}_1(\mathcal{G}) = \mathcal{N}(w_0, \mathrm{X}_1(\mathcal{G})) + \mathcal{N}(w_1, \mathrm{X}_2(\mathcal{G})) + \mathcal{N}(w_2, \mathrm{X}_0(\mathcal{G}))$$

$$\mathcal{V}_2(\mathcal{G}) = \mathcal{N}(w_0, \mathrm{X}_2(\mathcal{G})) + \mathcal{N}(w_1, \mathrm{X}_0(\mathcal{G})) + \mathcal{N}(w_2, \mathrm{X}_1(\mathcal{G}))$$

Les mots des codes $X_0(\mathcal{G})$, $X_1(\mathcal{G})$ et $X_0(\mathcal{G})$ sont associés préférentiellement aux phases 0 (1 et 2 resp.). Par conséquent, la phase estimée de w_0 est la phase 0 (1 et 2 resp.) si la valeur de $\mathcal{V}_0(\mathcal{G})$ ($\mathcal{V}_1(\mathcal{G})$ et $\mathcal{V}_2(\mathcal{G})$ resp.) est forte

$$\text{La phase proposée } p' \text{ est telle que } \mathcal{V}_{p'}(\mathcal{G}) = MAX_{p'=0}^{2}\{\mathcal{V}_{p'}(\mathcal{G})\}$$

Pour chaque mot w extrait de séquences génétiques d'un génome, la phase calculée du mot est comparée à la phase réelle. La phase proposée est correcte lorsqu'elle est identique à la phase réelle et qu'il n'y a pas d'ambiguïté dans le choix de la valeur la plus forte de $\mathcal{V}(\mathcal{G})$ (les deux valeurs les plus fortes de $\mathcal{V}(\mathcal{G})$ sont différentes). La probabilité de retrouver la phase réelle d'un mot w est estimée pour des longueurs $|w|$ comprises entre 5 (1 trinucléotide dans chaque phase) et 50 nucléotides.

Pour chaque longueur $|w|$, plus de 35 millions de mots extraits de séquences provenant de 175 génomes de bactéries ont été examinés. Les résultats sont visibles sur la figure 4.1. La probabilité de retrouver correctement la phase varie entre 0.48 pour les mots de 5 nucléotides et 0.810 pour les mots de 50 nucléotides. A titre de comparaison, pour les mots de longueur 13, ce qui correspond à la plus grande fenêtre possible pour des codes circulaires, la probabilité de retrouver la phase dans des séquences génétiques réelles par cette méthode est de 0.586. La probabilité de retrouver la phase par un choix purement aléatoirement est de $\frac{1}{3}$.

La méthode a été également appliquée par génome. Les résultats montrent des variations en fonction de la "force" des codes circulaires, c'est à dire en fonction de $\mathcal{F}(\mathrm{S})$. Par exemple, la probabilité de redécouvrir la phase varie entre 0.404 et 0.640 pour les mots de longueur 5 et varie entre 0.619 et 0.978 pour les mots de longueur 50.

Finalement, ces probabilités basées sur les codes C^3 de procaryotes sont comparées à celle obtenues en prenant en compte les 20 mots les plus fréquents par phase. L'utilisation des trinucléotides les plus fréquents devrait a priori donner de meilleurs résultats. Pourtant pour des mots courts (≤ 25 nucléotides), il est plus avantageux d'utiliser les codes C^3 (fig. 4.1). Les codes circulaires ne contiennent pas de mots permutés, les propriétés des codes circulaires deviennent moins utiles pour des mots de plus de 25 nucléotides.

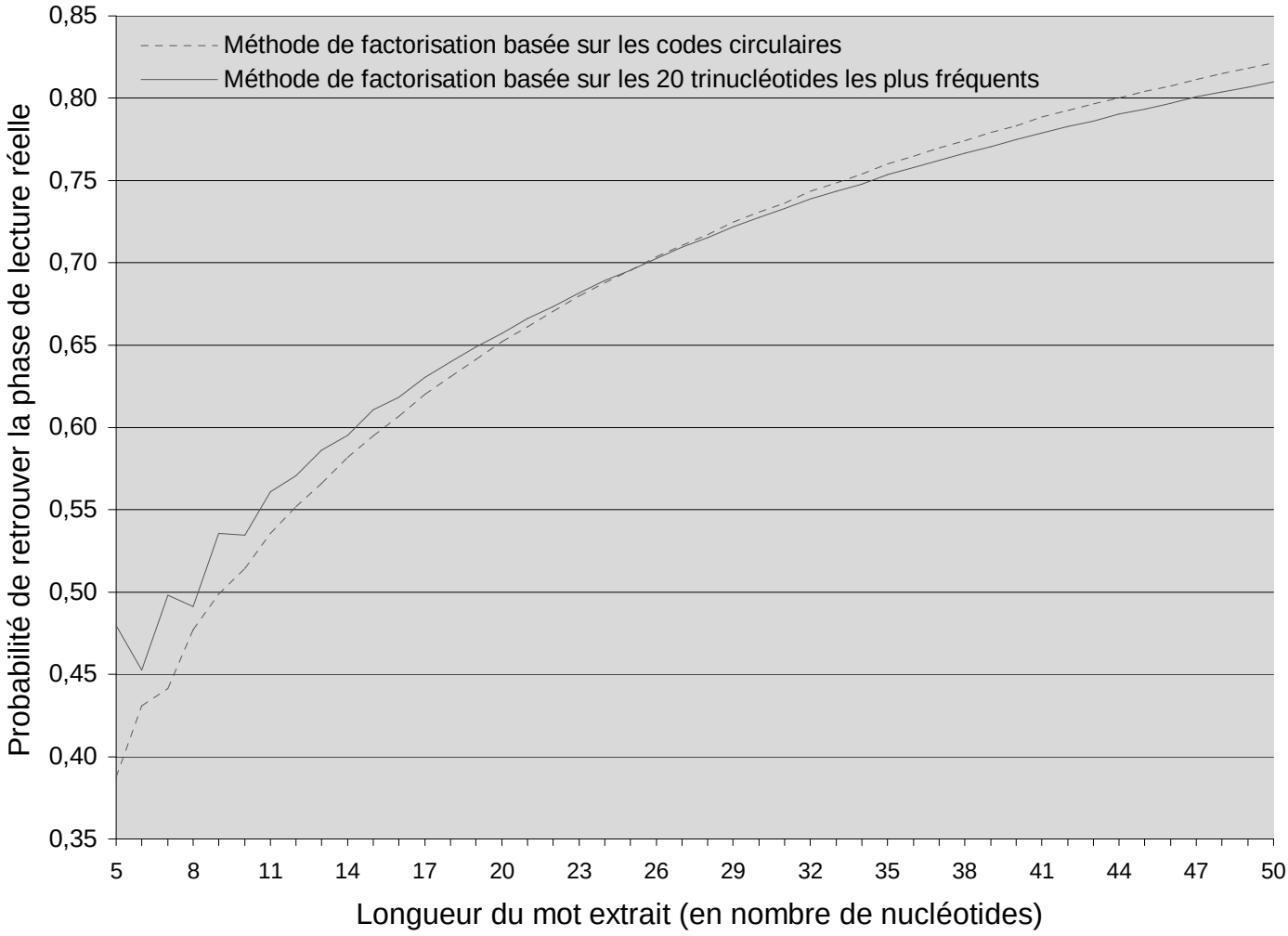

Figure 4.1 Probabilité de retrouver la phase réelle d'un mot extrait à une position quelconque de gènes choisis aléatoirement parmi les 191 génomes de procaryotes en fonction de la longueur des mots en nombre de nucléotides. Deux méthodes de factorisation sont étudiées. La première se base sur les ensembles de trinucléotides obtenus par la méthode FPTF (ligne continue) et la seconde sur les ensembles de 20 codons les plus fréquents (ligne discontinue).

4.6 Comparaison au cas aléatoire

La probabilité d'occurrence des codes circulaires $\mathcal{C}^3$ est très faible $221544/3^{20} \simeq 6.3 \times 10^{-5}$. Cette probabilité est obtenue en calculant le nombre de codes $\mathcal{C}^3$ (221544) parmi les 3^{20} ensembles de trinucléotides potentiellement générés par la méthode FPTF.

De plus, la signification statistique des $3 \times 175 = 525$ codes circulaires de bactéries $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$ associés aux 3 phases des gènes des 175 génomes est évaluée en étant comparée aux ensembles identifiés par la méthode FPTF pour des séquences générées aléatoirement.

Pour chacun des génomes de bactéries $\mathcal{G}$ est généré plusieurs génomes aléatoires. Un génome aléatoire $\mathcal{R}$ a un nombre de gènes ainsi qu'une distribution des longueurs de gènes identiques à ceux du génome réel auquel il est associé. Les séquences aléatoires dont la distribution des nucléotides est équiprobable ne sont pas représentatives des séquences génétiques naturelles, les gènes présentant des biais dans l'utilisation des nucléotides. La valeur obtenue par la fonction $\mathcal{F}$ pour la méthode FPTF appliquée à des séquences aléatoires ne présentant pas de biais dans l'utilisation des nucléotides, dinucléotides ou trinucléotides est minimale (1/3, voir 5.3), les ensembles obtenus ne peuvent pas être significatifs. Afin d'obtenir des séquences plus proches des gènes réels, l'ensemble des séquences aléatoires $\mathcal{R}_N$ (resp. $\mathcal{R}_D$ et $\mathcal{R}_T$) est généré en respectant des contraintes sur la distribution de nucléotides (resp. dinucléotides ou trinucléotides).

Ainsi, les ensembles $\mathcal{R}_N$ (resp. $\mathcal{R}_D$ et $\mathcal{R}_T$) auront des distributions de nucléotides (resp. dinucléotides ou trinucléotides) similaires à celles des séquences codantes des génomes $\mathcal{G}$. Afin d'obtenir une plus grande diversité dans les séquences aléatoires potentiellement générées, les fréquences des n-nucléotides des séquences réels sont conservées mais elles sont mélangées et associées aléatoirement aux différents n-nucléotides.

Les ensembles $\mathcal{R}_N$, $\mathcal{R}_D$ et $\mathcal{R}_T$ contiennent chacun plus de 3000 génomes générés aléatoirement. Pour chaque génome d'un ensemble $\mathcal{R}$, les 3 ensembles de trinucléotides $\mathcal{X}_0$, $\mathcal{X}_1$ et $\mathcal{X}_2$ sont déterminés par la méthode FPTF. La longueur du plus grand code circulaire contenu dans chaque ensemble est calculée. Il est important de noter que, étant donné que pour un ensemble de longueur 20, il existe $\binom{20}{n}$ sous-ensembles de longueur n et que lorsqu'un ensemble n'est pas un code circulaire, tous ses sous-ensembles sont testés, la probabilité d'identifier des codes circulaires augmentent fortement lorsque n diminue. La longueur moyenne des codes circulaires identifiés pour chaque phase des génomes aléatoires

ainsi que la longueur du plus grand code $\mathcal{C}^3$ sont calculées (voir table 4.5). Ces valeurs sont comparées aux résultats obtenus pour les génomes de bactéries.

La longueur moyenne des codes circulaires , les longueurs moyennes des codes circulaires par phase ainsi que la longueur moyenne des codes $\mathcal{C}^3$ pour les génomes bactériens $\mathcal{G}$ et les génomes aléatoires $\mathcal{R}_N$, $\mathcal{R}_D$ et $\mathcal{R}_T$ sont dans la table 4.5. Pour les ensembles de génomes $\mathcal{G}$, $\mathcal{R}_N$, $\mathcal{R}_D$ et $\mathcal{R}_T$, les longueurs moyennes sont presque identiques pour chaque phase et sont sensiblement plus faibles que celles des génomes bactériens. La longueur moyenne des codes de bactéries est de 19.77 mots et n'est que de 17.45, 17.41 et 17.91 mots pour les ensembles $\mathcal{R}_N$, $\mathcal{R}_D$ et $\mathcal{R}_T$ respectivement. La différence est encore plus grande pour les longueurs moyennes des codes $\mathcal{C}^3$ 19.5 mots pour les bactéries et 15.78, 15.72 et 16.48 pour les ensembles $\mathcal{R}_N$, $\mathcal{R}_D$ et $\mathcal{R}_T$ respectivement.

Ces calculs montrent la différence significative entre les longueurs moyennes des codes circulaires des génomes de bactérie et celles des génomes aléatoires. Les codes circulaires bactériens sont proches de la maximalité (19.77).

	Longueur moyenne des codes circulaires (en phase 0, 1, 2 resp.)	Longueur moyenne $\mathcal{C}^3$
Génomes bactériens $\mathcal{G}$	19.77 (19.81, 19.79, 19.70)	19.5
$\mathcal{R}_N$	17.45 (17.42, 17.46, 17.74)	15.78
$\mathcal{R}_D$	17.41 (17.40, 17.42, 17.42)	15.72
$\mathcal{R}_T$	17.91 (18.15, 17.84, 17.83)	16.48

Table 4.5 : Longueur moyennes des codes circulaires dans les génomes de bactéries et dans des génomes aléatoires.

De nouveaux codes circulaires ont été identifiés dans les génomes de procaryotes. Il est intéressant d'étudier la relation pouvant exister entre ces nouveaux codes et le code commun. Dans le chapitre suivant, nous avons recherché et identifié une relation d'évolution entre certains de ces nouveaux codes (les codes des archaea) et le code commun $\mathcal{X}$.

Chapitre 5

Modèle d'évolution pour les codes circulaires

5.1 Présentation du modèle d'évolution

Un modèle d'évolution analytique basé sur des matrices de mutation 64×64 avec 6 paramètres de substitution associés aux transitions et aux transversions dans les 3 sites des trinucléotides, a été développé. Il permet de déterminer à un moment t la probabilité exacte d'occurrence de trinucléotides mutant aléatoirement suivant ces 6 paramètres de substitutions. Une application de ce modèle a permis d'étudier l'évolution du code circulaire $\mathcal{X}$ des eucaryotes et des procaryotes (4.3) et des 15 codes circulaires identifiés dans plusieurs génomes d'archaea [FM03]. Dans les gènes, le signal statistique des codes circulaires se superpose à l'information du code génétique. Le code $\mathcal{X}$ est considéré comme primitif car son signal est présent dans différentes populations de gènes et il est auto-complémentaire. Aucun code circulaire d'archaea n'est autocomplémentaire.
Le modèle proposé ici permet de rechercher si des valeurs des 6 paramètres de mutation permettent de dériver les codes circulaires des archaea du code circulaire $\mathcal{X}$ supposé primitif. Les taux de substitutions qui seront obtenus pour les différents organismes permettent de regrouper l'évolution des 15 codes archaea en 3 classes principales. Dans pratiquement tous les cas, les mutations observées sont cohérentes avec la dégénérescence du code génétique, les substitutions étant plus fréquentes dans le 3^{eme} site des trinucléotides et les transitions plus fréquentes que les transversions en général.

L'observation d'un ensemble $\mathcal{X}$ de trinucléotides préférentiels dans divers gènes des eucaryotes et des procaryotes est à la base du développement du modèle d'évolution proposé. En effet, si un ensemble de trinucléotides préférentiels apparaît avec une fréquence plus élevée que d'autres ensembles de trinucléotides choisis aléatoirement même après des mutations aléatoires, une hypothèse réaliste consiste à supposer que cet ensemble avait dans le passé une fréquence d'occurrence dans les gènes plus élevée que celle observé actuellement. En conséquence, les trinucléotides de $\mathcal{X}$ auraient un rôle plus important dans les gènes "primitifs".

Le modèle d'évolution se base sur 2 processus : une construction avec un mélange aléatoire des 20 trinucléotides de $\mathcal{X}$, chacun ayant une fréquence d'occurrence identique (1/20), puis un processus d'évolution avec des taux de substitution aléatoire. Six paramètres de substitution sont considérés. Ils sont associés aux transitions et aux transversions sur les trois sites. Une transition est une substitution d'une purine $\{A, G\}$ par une autre purine ou d'une pyrimidine $\{C, T\}$ par une autre pyrimidine. Une transversion est la substitution d'une purine par une pyrimidine ou d'une pyrimidine par une purine. Les différents taux

de mutation sont notés a (resp. c, e) correspondant aux taux de transition pour le premier (resp. second et troisième) site des trinucléotides et b (resp. d, f) associé aux taux de transversion pour le premier (resp. second et troisième) site des trinucléotides.

Le modèle mathématique d'évolution est basé sur une matrice de mutation de trinucléotide 64×64 à 6 paramètres. Il étend ainsi les modèles d'évolution classique basés sur une matrice de mutation des nucléotides 4×4 avec 1 paramètres [Juk69] ou 2 paramètres [Kim80](transitions et transversions) ou sur une matrice de mutation de trinucléotide 64×64 à 3 paramètres [AM98].

5.2 Codes des archaea

Les archaea correspondent au dernier grand domaine de la vie identifié [Woe77]. Ce sont des organismes monocellulaires de type procaryote, ils sont cependant distincts des bactéries. En effet, au niveau moléculaire, les archaea (aussi appelés archées ou archéobactéries) possèdent à la fois certaines caractéristiques des eucaryotes et des procaryotes. Elles ont également des propriétés qui leur sont propres ([Ber00], [Woe00], [For01]). Elles possèdent un mode d'organisation cellulaire de type procaryote mais présentent de nombreuses similarités avec les eucaryotes dans leur processus de réplication, de transcription ou de traduction. Le domaine des archaea est connu pour contenir un certain nombre d'organismes extrémophiles, c'est à dire adaptés à la vie dans des environnements extrêmes, comme par exemple des milieux à très forte ou très basse température, fortement acide, etc. Les archaea sont considérées comme majoritaires sur les bactéries dans la plupart des milieux hostiles.

La méthode FPTF a permis d'identifier 15 codes circulaires C^3 associés à 16 génomes d'archaea. Ces codes sont tous différents du code $\mathcal{X}$ et ne possèdent pas la propriété d'autocomplémentarité. Les propriétés de ces codes sont données dans le chapitre 2. Afin de quantifier la prédominance des codes d'archaea X sur le code commun $\mathcal{X}$, une comparaison de la prévalence des différents codes dans les génomes d'archaea est effectuée. Les probabilités suivantes sont définies. Soit $P_i(G)$ la probabilité d'occurrence d'un trinucléotide i, $i \in \{1, \ldots, 64\}$ représentant les 64 trinucléotides $\mathbb{T} = \{\mathrm{AAA}, \mathrm{AAC}, \ldots, \mathrm{TTG}, \mathrm{TTT}\}$, dans les gènes du génome G. Comme les trinucléotides $\widetilde{\mathbb{T}} = \{\mathrm{AAA}, \mathrm{CCC}, \mathrm{GGG}, \mathrm{TTT}\}$ ne peuvent appartenir à un code circulaire, ils ne sont donc pas pris en compte dans le calcul de la probabilité d'occurrence d'un code circulaire. La probabilité d'occurrence $P(X, G)$ d'un code X dans un génome G est donc normalisée

$$P(X, G) = \frac{\sum\limits_{i \in X} P_i(G)}{\sum\limits_{i \in \mathbb{T} - \widetilde{\mathbb{T}}} P_i(G)}.$$

Ainsi, dans un génome d'archaea G, la différence de probabilité $\Pr(X, \mathcal{X}, G)$ évalue l'occurrence d'un code X comparée à celle du code $\mathcal{X}$ tel que :

$$\Pr(X, \mathcal{X}, G) = P(X, G) - P(\mathcal{X}, G). \tag{5.1}$$

Dans les 16 génomes d'archaea, $\Pr(X, \mathcal{X}, G) > 0$, $\Pr(MSA, \mathcal{X}, G_{MSA}) = 1.22\%$ étant la plus faible valeur (Table 5.1.). Par conséquent, les 15 codes d'archaea apparaissent avec

une fréquence plus forte que le code $\mathcal{X}$ dans les gènes des génomes d'archaea.

G	AG	AP	HB	MC	MP	MSA	MSM	MT	PB	PCA	PCF	PCH	SLS	SLT	TPA	TPV
Pr	4.80	6.61	3.45	9.64	5.23	1.22	2.54	2.70	5.63	7.32	7.72	5.08	7.37	11.43	5.46	5.51

Table 5.1. Différence de probabilité $\Pr(X, \mathcal{X}, G) = P(X, G) - P(\mathcal{X}, G)$ (en %) entre un code circulaire d'archaea X et le code commun $\mathcal{X}$ dans les 16 génomes d'archaea G.

Le modèle analytique d'évolution développé dans la prochaine section montrera qu'il est possible pour certain taux de mutations de dériver les codes d'archaea du code commun $\mathcal{X}$.

5.3 Modèle mathématique

Le modèle mathématique proposé détermine une solution analytique des probabilités d'occurrence du code circulaire $\mathcal{X}$ et des 15 codes circulaires X en fonction du temps d'évolution t et des 6 paramètres de substitution a, b, c, d, e et f. Cette approche stochastique s'appuie sur un modèle physique d'évolution des gènes basé sur la construction de gènes primitifs puis d'un processus de mutation aléatoire.

Afin d'obtenir des résultats avec une bonne approximation, une importante population de séquences doit être simulée et analysée statistiquement.

Par convention, les indices $i, j \in \{1, ..., 64\}$ représentent les 64 trinucléotides $\mathbb{T}$ dans l'ordre alphabétique. La probabilité d'occurrence $P_i(t+dt)$ d'un trinucléotide i à un temps $t+dt$ est égale à la probabilité d'occurrence $P_i(t)$ de ce trinucléotide i au temps t moins la probabilité de substitution de ce trinucléotide i durant $[t, t+dt]$ et plus la probabilité qu'une substitution transforme le trinucléotide j, $j \neq i$, en un trinucléotide i durant $[t, t+dt]$

$$P_i(t+dt) = P_i(t) - \alpha dt P_i(t) + \alpha dt \sum_{j=1}^{64} P(j \to i)\, P_j(t) \tag{5.2}$$

où α est la probabilité qu'un trinucléotide subisse 1 substitution durant un intervalle de temps et où $P(j \to i)$ est la probabilité qu'une substitution transforme le trinucléotide j en un trinucléotide i. La probabilité $P(j \to i)$ est égale à 0 si la substitution est impossible (j et i différent de plus d'un nucléotide et sachant que dt est considéré comme suffisamment petit pour qu'un trinucléotide ne puisse pas muter successivement 2 fois durant dt) sinon elle est donnée en fonction des 6 taux de substitution a, b, c, d, e et f. Par exemple, pour le trinucléotide AAA, associé à $i = 1$, $P(\text{GAA}\to\text{AAA}) = a$, $P(\text{CAA}\to\text{AAA}) = P(\text{TAA}\to\text{AAA}) = b/2$, $P(\text{AGA}\to\text{AAA}) = c$, $P(\text{ACA}\to\text{AAA}) = P(\text{ATA}\to\text{AAA}) = d/2$, $P(\text{AAG}\to\text{AAA}) = e$, $P(\text{AAC}\to\text{AAA}) = P(\text{AAT}\to\text{AAA}) = f/2$ et $P(j \to \text{AAA}) = 0$ avec $j \notin \{\text{AAC,AAG,AAT,ACA,AGA,ATA,CAA,GAA,TAA}\}$.

Pour une unité de temps appropriée, la probabilité α est égale à 1, c'est à dire qu'il existe une substitution par trinucléotide par unité de temps. Ainsi, la formule (5.2) devient

$$\frac{P_i(t+dt) - P_i(t)}{dt} \approx P_i'(t) = -P_i(t) + \sum_{j=1}^{64} P(j \to i)\, P_j(t). \tag{5.3}$$

En considérant le vecteur colonne $P(t) = (P_i(t))_{1 \leq i \leq 64}$ composé des 64 $P_i(t)$ et la matrice de mutation A $(64, 64)$ des 4096 probabilités de substitution de trinucléotide $P(j \to i)$,

l'équation différentielle (5.3) peut être représentée par la notation matricielle suivante

$$P'(t) = -P(t) + A \cdot P(t) = (A - I) \cdot P(t) \tag{5.4}$$

où I représente la matrice identité et le symbole $\cdot$, le produit matriciel.

La matrice carrée A $(64, 64)$ peut être définie par une matrice bloc carrée $(4, 4)$ dont les 4 éléments diagonaux sont formés par 4 sous-matrices carrées identiques B $(16, 16)$ et dont les 12 éléments non-diagonaux sont formés de 4 sous-matrices carrées aI $(16, 16)$ et de 8 sous-matrices carrées $(b/2)I$ $(16, 16)$ telle que

$$A = \left(\begin{array}{c|cccc} & 1\cdots16 & 17\cdots32 & 33\cdots48 & 49\cdots64 \\ \hline 1\cdots16 & B & (b/2)I & aI & (b/2)I \\ 17\cdots32 & (b/2)I & B & (b/2)I & aI \\ 33\cdots48 & aI & (b/2)I & B & (b/2)I \\ 49\cdots64 & (b/2)I & aI & (b/2)I & B \end{array}\right).$$

Les blocs indicés $\{1, ..., 16\}$, $\{17, ..., 32\}$, $\{33, ..., 48\}$ et $\{49, ..., 64\}$ sont associés aux trinucléotides {AAA,...,ATT}, {CAA,...,CTT}, {GAA,...,GTT} et {TAA,...,TTT} respectivement. La sous-matrice carrée B $(16, 16)$ peut aussi être définie par une matrice bloc carrée $(4, 4)$ dont les 4 éléments diagonaux sont formés de 4 sous-matrices carrées identiques C $(4, 4)$ et dont les 12 éléments sont formés de 4 sous-matrices carrées cI $(4, 4)$ et de 8 sous-matrices carrées $(d/2)I$ $(4, 4)$ tel que

$$B = \begin{pmatrix} C & (d/2)I & cI & (d/2)I \\ (d/2)I & C & (d/2)I & cI \\ cI & (d/2)I & C & (d/2)I \\ (d/2)I & cI & (d/2)I & C \end{pmatrix}.$$

Enfin, la sous-matrice carrée C $(4, 4)$ est égale à

$$C = \begin{pmatrix} 0 & f/2 & e & f/2 \\ f/2 & 0 & f/2 & e \\ e & f/2 & 0 & f/2 \\ f/2 & e & f/2 & 0 \end{pmatrix}.$$

La matrice A est stochastique quand $a + b + c + d + e + f = 1$.

L'équation différentielle () peut être écrite sous la forme suivante :

$$P'(t) = M \cdot P(t)$$

avec

$$M = A - I.$$

Comme les 6 paramètres de substitution sont réels, la matrice A est réelle. Elle est aussi symétrique par construction. Par conséquent, la matrice M est aussi réelle et symétrique. Il existe donc une matrice de vecteurs propres Q et une matrice diagonale D de valeurs propres λ_k de M ordonnée de la même manière que les colonnes des vecteurs propres de Q tel que $M = Q \cdot D \cdot Q^{-1}$. Alors,

$$P'(t) = Q \cdot D \cdot Q^{-1} \cdot P(t).$$

Cette équation a la solution classique suivante (voir [Lan05])

$$P(t) = Q \cdot e^{Dt} \cdot Q^{-1} \cdot P(0) \tag{5.5}$$

où e^{Dt} est la matrice diagonale des valeurs propres exponentielles $e^{\lambda_k t}$.

Les valeurs propres λ_k de M sont déduites des valeurs propres μ_k de A tels que $\lambda_k = \mu_k - 1$. Les valeurs propres μ_k de A peuvent être obtenues en déterminant les racines de l'équation caractéristique $\det(A - \mu I) = 0$ de A en se servant des propriétés des matrice blocs. Par conséquent, après combinaison linéaire, le déterminant $\det(A - \mu I)$ est égal à

$$\det(A-\mu I) = \det\left(B - (-a+b+\mu)\, I\right) \times \det\left(B - (-a-b+\mu)\, I\right) \times \left[\det\left(B - (a+\mu)\, I\right)\right]^2. \tag{5.6}$$

Comme la matrice B a une structure en bloc similaire à celle de la matrice A, le déterminant $\det(B - \nu I)$ peut être facilement déduit de $\det(A - \mu I)$

$$\det(B-\nu I) = \det\left(C - (-c+d+\nu)\, I\right) \times \det\left(C - (-c-d+\nu)\, I\right) \times \left[\det\left(C - (c+\nu)\, I\right)\right]^2.$$

Par conséquent, en substituant dans [] $\nu = -a+b+\mu$, $\nu = -a-b+\mu$ et $\nu = a+\mu$, le déterminant $\det(A-\mu I)$ devient

$$\begin{aligned}\det(A-\mu I) &= \det\left(C-(-a+b-c+d+\mu)\,I\right)\times\det\left(C-(-a+b-c-d+\mu)\,I\right) \quad (5.7)\\ &\times\det\left(C-(-a-b-c+d+\mu)\,I\right)\times\det\left(C-(-a-b-c-d+\mu)\,I\right)\\ &\times\left[\det\left(C-(-a+b+c+\mu)\,I\right)\right]^2\times\left[\det\left(C-(-a-b+c+\mu)\,I\right)\right]^2\\ &\times\left[\det\left(C-(a-c+d+\mu)\,I\right)\right]^2\times\left[\det\left(C-(a-c-d+\mu)\,I\right)\right]^2\\ &\times\left[\det\left(C-(a+c+\mu)\,I\right)\right]^4.\end{aligned}$$

Après combinaison linéaire, le déterminant $\det(C-X.I)$ est égal à

$$\det(C-X.I) = (e-f-X)(e+f-X)(-e-X)^2.$$

En substituant dans (5.7) $X = -a+b-c+d+\mu$, $X = -a+b-c-d+\mu$, $\mathcal{X} = -a-b-c+d+\mu$, $X = -a-b-c-d+\mu$, $X = -a+b+c+\mu$, $X = -a-b+c+\mu$, $X = a-c+d+\mu$, $X = a-c-d+\mu$ ou $X = a+c+\mu$, le déterminant $\det(A-\mu I)$ est obtenu

$$\begin{aligned}\det(A-\mu I) &= (a+b+c+d+e+f-\mu)(a+b+c+d+e-f-\mu)(a+b+c-d+e+f-\mu)\\ &\times(a+b+c-d+e-f-\mu)(a-b+c+d+e+f-\mu)(a-b+c+d+e-f-\mu)\\ &\times(a-b+c-d+e+f-\mu)(a-b+c-d+e-f-\mu)\\ &\times(a+b+c+d-e-\mu)^2(a+b+c-d-e-\mu)^2(a+b-c+e+f-\mu)^2\\ &\times(a+b-c+e-f-\mu)^2(a-b+c+d-e-\mu)^2(a-b+c-d-e-\mu)^2\\ &\times(a-b-c+e+f-\mu)^2(a-b-c+e-f-\mu)^2(-a+c+d+e+f-\mu)^2\\ &\times(-a+c+d+e-f-\mu)^2(-a+c-d+e+f-\mu)^2(-a+c-d+e-f-\mu)^2\\ &\times(a+b-c-e-\mu)^4(a-b-c-e-\mu)^4(-a+c+d-e-\mu)^4\\ &\times(-a+c-d-e-\mu)^4(-a-c+e+f-\mu)^4(-a-c+e-f-\mu)^4\\ &\times(-a-c-e-\mu)^8.\end{aligned}$$

Il existe donc 27 valeurs propres λ_k de M. Il y a 8 valeurs propres de multiplicité algébrique 1 : $\lambda_1 = -1+a+b+c+d+e+f$, $\lambda_2 = -1+a+b+c+d+e-f$, $\lambda_3 = -1+a+b+c-d+e+f$, $\lambda_4 = -1+a+b+c-d+e-f$, $\lambda_5 = -1+a-b+c+d+e+f$, $\lambda_6 = -1+a-b+c+d+e-f$, $\lambda_7 = -1+a-b+c-d+e+f$ et $\lambda_8 = -1+a-b+c-d+e-f$. Il y a 12 valeurs propres de multiplicité algébrique 2 : $\lambda_9 = -1+a+b+c+d-e$, $\lambda_{10} = -1+a+b+c-d-e$, $\lambda_{11} = -1+a+b-c+e+f$, $\lambda_{12} = -1+a+b-c+e-f$,

$\lambda_{13} = -1 + a - b + c + d - e$, $\lambda_{14} = -1 + a - b + c - d - e$, $\lambda_{15} = -1 + a - b - c + e + f$, $\lambda_{16} = -1 + a - b - c + e - f$, $\lambda_{17} = -1 - a + c + d + e + f$, $\lambda_{18} = -1 - a + c + d + e - f$, $\lambda_{19} = -1 - a + c - d + e + f$ et $\lambda_{20} = -1 - a + c - d + e - f$. Il y a 6 valeurs propres de multiplicité algébrique 4 : $\lambda_{21} = -1 + a + b - c - e$, $\lambda_{22} = -1 + a - b - c - e$, $\lambda_{23} = -1 - a + c + d - e$, $\lambda_{24} = -1 - a + c - d - e$, $\lambda_{25} = -1 - a - c + e + f$, $\lambda_{26} = -1 - a - c + e - f$. Il y a une 1 valeur propres de multiplicité algébrique 8 : $\lambda_{27} = -1 - a - c - e$.

Les 64 vecteurs propres de M associés à ces 27 valeurs propres λ_k obtenus par calcul formel sont indépendants de a, b, c, d, e et f.

Le mélange indépendant et équiprobable (1/20) des 20 trinucléotides de $\mathcal{X}$ donne le vecteur initial suivant
$P(0)$ =[0, 1/20, 0, 1/20, 0, 1/20, 0, 0, 0, 0, 0, 0, 0, 1/20, 0, 1/20, 0, 0, 1/20, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1/20, 1/20, 0, 1/20, 1/20, 1/20, 1/20, 0, 1/20, 0, 0, 0, 1/20, 0, 1/20, 1/20, 1/20, 0, 1/20, 0, 1/20, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1/20, 0, 0]

La formule avec 64 probabilités de trinucléotides $P_j(0)$ avant le processus de substitution ($t = 0$), la matrice diagonale e^{Dt} des valeurs propres exponentielles $e^{\lambda_k t}$ de M, sa matrice de vecteurs propres Q et son inverse Q^{-1}, permettent de déterminer les 64 probabilités de trinucléotides $P_i(t)$ après t substitutions en fonction des 6 paramètres de substitutions a, b, c, d, e et f.

Comme le code circulaire, X ne peut pas contenir un trinucléotide $\widetilde{\mathbb{T}}$ par définition, la probabilité d'occurrence $P(X,t)$ d'un code circulaire X à l'étape de substitution, est

$$P(X,t) = \frac{\sum\limits_{i \in X} P_i(t)}{\sum\limits_{i \in \mathbb{T} - \widetilde{\mathbb{T}}} P_i(t)}.$$

Finalement, la formule analytique d'évolution $P(\mathcal{X},t)$ du code circulaire commun $\mathcal{X}$ en fonction des 6 taux de substitution a, b, c, d, e et f associés aux transitions et aux transversions dans les 3 sites des trinucléotides et peut être exprimée en fonction des valeurs propres λ_k de M

$$P(\mathcal{X},t) = \frac{1}{2D}(100 + 25e^{\lambda_2 t} + e^{\lambda_4 t} + 25e^{\lambda_5 t} + 16e^{\lambda_6 t} + e^{\lambda_7 t} + 13e^{\lambda_9 t} + 5e^{\lambda_{10} t} + 36e^{\lambda_{11} t} \tag{5.8}$$

$$+ 2e^{\lambda_{12} t} + 5e^{\lambda_{13} t} + e^{\lambda_{14} t} + 2e^{\lambda_{15} t} + 13e^{\lambda_{17} t} + 5e^{\lambda_{18} t} + 5e^{\lambda_{19} t} + e^{\lambda_{20} t} + 2e^{\lambda_{21} t}$$

$$+ 22e^{\lambda_{22} t} + 6e^{\lambda_{23} t} + 2e^{\lambda_{24} t} + 2e^{\lambda_{25} t} + 22e^{\lambda_{26} t} + 8e^{\lambda_{27} t}\Big)$$

avec le dénominateur D

$$D = 150 - e^{\lambda_4 t} + 4e^{\lambda_6 t} - e^{\lambda_7 t} + e^{\lambda_{21} t} + 3e^{\lambda_{22} t} + 2e^{\lambda_{23} t} - 2e^{\lambda_{24} t} + e^{\lambda_{25} t} + 3e^{\lambda_{26} t}. \tag{5.9}$$

Les formules analytiques d'évolution $P(X,t)$ des 15 codes circulaires d'archaea X sont donnés dans l'annexe 3.

Propriété 1 : La probabilité initiale $P(\mathcal{X},0)$ (resp. $P(X,0)$) du code $\mathcal{X}$ (resp. d'un code d'archaea X) au temps $t = 0$ peut évidemment être obtenue à partir de la solution analytique $P(\mathcal{X},t)$ (resp. $P(X,t)$) avec $t = 0$ ou aussi par un simple calcul de probabilité.

La probabilité $P(\mathcal{X},0)$ est égale à 1 car les gènes primitifs dans ce modèle d'évolution sont générés par le code $\mathcal{X}$ (20 parmi 20 trinucléotides).

La probabilité $P(X,0)$ est aussi égale au ratio du nombre de trinucléotide commun à $\mathcal{X}$ et X sur 20. Ces probabilités $P(X,0)$ sont données dans la table 5.2.

X	AG/AP	HB	MC	MP	MSA	MSM	MT	PB	PCA	PCF	PCH	SLS	SLT	TPA	TPV
$P(X,0)$	$\frac{7}{10}$	$\frac{7}{10}$	$\frac{1}{2}$	$\frac{7}{10}$	$\frac{3}{4}$	$\frac{7}{10}$	$\frac{3}{4}$	$\frac{3}{4}$	$\frac{7}{10}$	$\frac{3}{5}$	$\frac{3}{5}$	$\frac{11}{20}$	$\frac{1}{2}$	$\frac{7}{10}$	$\frac{7}{10}$

Table 5.2. Probabilités initiales $P(X,0)$ des 15 codes circulaires d'archaea X.

Propriété 2 : La probabilité $P(\mathcal{X},t)$ (resp. $P(X,t)$) du code $\mathcal{X}$ (resp. un code d'archaea X) au temps limite $t \to \infty$ peut être obtenue (trivialement) par l'étude de la limite de la fonction ou par un simple calcul de probabilité.

Quelque soit $a,b,c,d,e,f \in]0,1[$ tel que $a+b+c+d+e+f = 1$, $\lim_{t\to\infty} P(\mathcal{X},t) = \lim_{t\to\infty} P(X,t) = 1/3$. En effet, les 6 substitutions des 20 trinucléotides de $\mathcal{X}$, ou de X, génèrent les 44 autres trinucléotides. Quand $t \to \infty$, les 64 trinucléotides $\mathbb{T}$ apparaissent avec la même probabilité et par conséquent, les probabilités de $\mathcal{X}$ et de X sont égales à $20/60 = 1/3$ (les 4 trinucléotides $\widetilde{\mathbb{T}}$ n'étant pas considérés).

Propriété 3 : Quand une ou plusieurs substitutions ont un taux égal à 0, certains trinucléotides peuvent être générés de façon non équiprobable ou peuvent ne pas être

générés et $\lim_{t\to\infty} P(\mathcal{X},t) \neq 1/3$, ou $\lim_{t\to\infty} P(\mathcal{X},t) \neq 1/3$. Par exemple, nous pouvons expliquer par un simple calcul de probabilité que $\lim_{t\to\infty} P(\mathcal{X},t) = 5/12$ quand $b = 0$. Le code $\mathcal{X}$ a 20 trinucléotides dont 15 trinucléotides commencent par une purine R. Les trinucléotides commençant par une base purine forment le sous-ensemble $\mathcal{X}_R$. L'ensemble $\mathcal{X}$ a aussi 5 trinucléotides commençant par une pyrimidine, ces trinucléotides formant le sous-ensemble $\mathcal{X}_Y$. On a alors $\mathcal{X} = \mathcal{X}_R \cup \mathcal{X}_Y$. Chaque $w \in \mathcal{X}$ apparaît avec la même probabilité $P(w) = 1/20$. Comme des bases puriques et des bases pyrimidiques existent dans le 1^{er} site des trinucléotides de $\mathcal{X}$ et comme les transitions et les transversions sont possibles dans le $2^{ème}$ et $3^{ème}$ sites de $\mathcal{X}$ $(c, d, e, f > 0)$, les 64 trinucléotides $\mathbb{T}$ sont générés au cours du processus d'évolution. Parmi ces 64 trinucléotides $\mathbb{T}$, soit $\mathbb{T}_R$ l'ensemble des 32 trinucléotides commençant par une purine et $\mathbb{T}_Y$, l'ensemble des 32 trinucléotides commençant par une pyrimidine, i.e. $\mathbb{T} = \mathbb{T}_R \cup \mathbb{T}_Y$. Comme les transitions sont possibles dans le 1^{er} site de $\mathcal{X}$ $(a > 0)$ mais pas les transversions $(b = 0)$, l'ensemble des trinucléotides $\mathbb{T}_R$ peut seulement être généré à partir de $\mathcal{X}_R$. Quand $t \to \infty$, les trinucléotides de $\mathcal{X}_R$ et de $\mathbb{T}_R$ apparaissent avec la même probabilité $P(w,t) = (15/20)/32 = 3/128$, $w \in \mathcal{X}_R$. De la même façon, lorsque $t \to \infty$, les trinucléotides de $\mathcal{X}_Y$ et $\mathbb{T}_Y$ apparaissent avec la même probabilité $P(w) = (5/20)/32 = 1/128$, $w \in \mathcal{X}_Y$. Les trinucléotides AAA et GGG (resp. CCC et TTT) appartiennent à $\mathbb{T}_R$ (resp. $\mathbb{T}_Y$). De plus, quand $t \to \infty$, les trinucléotides $\widetilde{\mathbb{T}}$ apparaissent avec la même probabilité $P(w) = (6+2)/128 = 1/16$, $w \in \widetilde{\mathbb{T}}$. Par conséquent, la limite $\lim_{t\to\infty} P(\mathcal{X},t)$ est égale à

$$\lim_{t\to\infty} P(\mathcal{X},t) = \frac{\sum_{w\in\mathcal{X}_R\cup\mathcal{X}_Y} \lim_{t\to\infty} P(w,t)}{1 - \lim_{\substack{t\to\infty \\ w\in\widetilde{\mathbb{T}}}} P(w,t)} = \frac{\frac{45+5}{128}}{1-\frac{1}{16}} = \frac{5}{12}.$$

Propriété 4 : La formule analytique d'évolution $Q(\mathcal{X},t)$ du code circulaire commun $\mathcal{X}$ en fonction des 3 taux de substitutions p, q et r associés respectivement aux 3 sites des trinucléotide, est un cas particulier de $P(\mathcal{X},t)$ avec $a = p/3$, $b = 2p/3$, $c = q/3$, $d = 2q/3$, $e = r/3$ et $f = 2r/3$

$$Q(\mathcal{X},t) = \frac{1}{2\mathcal{D}}\Big(50 + 28e^{-\frac{4}{3}t} + 5e^{-\frac{4}{3}(1-p)t} + 16e^{-\frac{4}{3}(1-q)t} + 19e^{-\frac{4}{3}(1-p-q)t} + 5e^{-\frac{4}{3}(1-r)t} + 18e^{-\frac{4}{3}(1-p-r)t} + 19e^{-\frac{4}{3}(1-q-r)t}\Big)$$

avec le dénominateur $\mathcal{D}$

$$\mathcal{D} = 75 + 2e^{-\frac{4}{3}t} + 3e^{-\frac{4}{3}(1-q)t}.$$

5.4 Evolution des codes d'archaea

Les 15 codes d'archaea X ont des probabilités initiales $P(X,0)$ variant entre 0.5 et 0.75, les 2 codes MC et SLT ont les probabilités les plus faibles, et les 3 codes MSA, MT et PB ont les probabilités les plus fortes (Table 5.2.). Toutes ces 15 probabilités $P(X,0)$ sont significativement inférieures aux probabilités initiales $P(\mathcal{X},0)=1$ du code circulaire commun $\mathcal{X}$. A priori, il semble donc difficile d'obtenir un processus de mutations aléatoires faisant dériver au cours de l'évolution les codes X du code commun $\mathcal{X}$, c'est à dire de faire décroître la courbe de probabilité $\mathcal{X}$ plus rapidement que celles des codes X de façon à ce que les code X apparaissent alors avec des probabilités plus grandes que $\mathcal{X}$.

Le modèle stochastique développé permet de rechercher des différences de probabilités positives entre X et $\mathcal{X}$

$$\Pr(X,\mathcal{X},t)=P(X,t)-P(\mathcal{X},t)>k \tag{5.10}$$

k étant choisi égal à 0.5% pour que la différence soit suffisamment significative. Chaque taux de substitution a, b, c, d, e et f varie dans l'intervalle $[0,1]$ avec un pas de 1% tel que la somme des probabilités soit égale à 1, et que t appartient à l'intervalle $[0,15]$.

Tous les codes d'archaea X peuvent dérivés par substitutions aléatoires du code commun $\mathcal{X}$. Ainsi, la différence $\Pr(X,\mathcal{X},t)$ est positive pour tous les codes X pour des valeurs de paramètres de substitution. La Table 5.3. donne les barycentres des espaces de solution (non indiqué) des 6 taux de substitutions a, b, c, d, e et f pour les 15 codes d'archaea. Le barycentre des taux permet une classification des 15 codes archaea conformément au modèle d'évolution avec 6 paramètres. Trois classes principales peuvent être obtenues, en fonction de valeurs faibles des taux de substitutions (Table 5.3.) :

(i) la classe $\mathcal{C}_r$ avec un taux de substitution faible pour le $3^{ème}$ site ($e<1\%$ et $f<10\%$, et $r<10\%$) contient le code TPA,

(ii) la classe $\mathcal{C}_q$ avec un taux de substitution faible pour le $2^{ème}$ site ($c<10\%$ et $d<10\%$, et $q \lessapprox 15\%$) contient les codes HB et MP,

(iii) la classe $\mathcal{C}_b$ avec un taux de substitution faible pour le 1^{er} site ($b<5\%$) peut être subdivisé en 5 sous-classes suivant les valeurs de b :

(iiia) la classe $\mathcal{C}_{b_1}$ contient les codes AG/AP et PB avec $b\approx 2\%$,

(iiib) la classe $\mathcal{C}_{b_2}$ contient les codes MC, PCA, SLS et SLT avec $b\approx 3\%$,

(iiic) la classe $\mathcal{C}_{b_3}$ contient les codes MT, PCF et PCH avec $b\approx 3.5\%$

(iiid) la classe $\mathcal{C}_{b_4}$ contient le code TPV avec $b\approx 4\%$

(iiie) la classe $\mathcal{C}_{b_5}$ contient les codes MSA et MSM avec $b \approx 4.5\%$.

X	a	b	c	d	e	f	$p = a + b$	$q = c + d$	$r = e + f$
AG/AP	24.9	1.9	14.5	18.8	13.4	26.5	26.8	33.3	39.9
HB	19.3	34.9	8.0	9.3	7.5	21.0	54.2	17.3	28.5
MC	10.5	2.9	20.6	20.6	17.0	28.4	13.4	41.2	45.4
MP	20.4	30.9	2.2	4.7	12.3	29.5	51.3	6.9	41.8
MSA	14.5	4.5	18.3	19.1	17.6	26.0	19.0	37.4	43.6
MSM	15.8	4.5	18.9	18.9	18.1	23.8	20.3	37.8	41.9
MT	17.8	3.5	18.0	19.9	16.0	24.8	21.3	37.9	40.8
PB	19.9	1.8	14.7	20.0	14.4	29.2	21.7	34.7	43.6
PCA	17.5	3.1	17.4	19.8	16.2	26.0	20.6	37.2	42.2
PCF	10.2	3.5	18.6	21.0	17.0	29.7	13.7	39.6	46.7
PCH	10.6	3.4	18.8	20.5	16.8	29.9	14.0	39.3	46.7
SLS	10.6	2.9	20.3	20.4	17.2	28.6	13.5	40.7	45.8
SLT	10.6	3.0	21.2	19.9	17.6	27.7	13.6	41.1	45.3
TPA	19.9	31.6	16.4	22.6	0.3	9.2	51.5	39.0	9.5
TPV	14.3	4.0	19.9	19.1	18.6	24.1	18.3	39.0	42.7

Table 5.3. Barycentres des taux de substitution (en %) de l'espace de solution pour les 15 codes d'archaea X tels que chaque code X apparaît avec une probabilité supérieure à celle de $\mathcal{X}$ (équation (5.10)).

L'existence d'une différence positive ne simule pas complètement la réalité. Une propriété plus forte est étudiée avec k égal à $\Pr(X, \mathcal{X}, G)$ ((5.1) et Table 5.3.), c'est à dire en cherchant une différence de probabilité entre chaque code d'archaea X et le code $\mathcal{X}$ qui soit supérieure à celle observée pour chaque génome

$$\Pr(X, \mathcal{X}, t) = P(X, t) - P(\mathcal{X}, t) > \Pr(X, \mathcal{X}, G) \tag{5.11}$$

Trois applications de ce modèle sont fortement corrélées aux codes d'archaea MSA, MSM et MT (Table 5.4.).

X	$\Pr(X, \mathcal{X}, G)$	a	b	c	d	e	f	Figure
MSA	1.22	13.8	2.8	18.3	19.9	17.6	27.6	1
MSM	2.54	15.0	1.4	19.0	20.3	18.3	26.0	2
MT	2.70	17.9	0.02	17.2	21.9	16.3	26.7	3

Table 5.4. Barycentres des taux de substitution (en %) de l'espace des solutions pour les 3 codes d'archaea $X = \{MSA, MSM, MT\}$ tels que chaque code X a une différence de probabilité

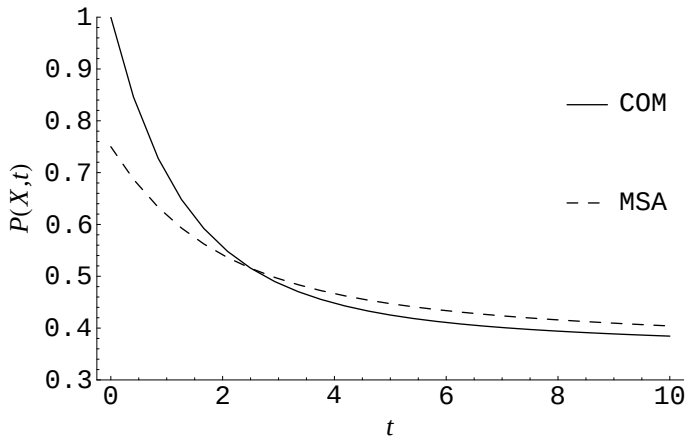

FIG. 5.1 – Evolution du code circulaire commun $\mathcal{X}$ et du code circulaire d'archaea MSA dans son barycentre des taux de substitution (en %) : $a = 13.8$, $b = 2.8$, $c = 18.3$, $d = 19.9$, $e = 17.6$ et $f = 27.6$ (Table 5.4.). La courbe $P(MSA, t)$ croise $P(\mathcal{X}, t)$ à $t_c \approx 2.51$ et est corrélée aux gènes réels des génomes d'archaea MSA à $t_a \approx 3.24$.

d'occurrence avec le code commun $\mathcal{X}$ plus élevée que celle observée dans son génome (équation (5.11) et $\Pr(X, \mathcal{X}, G)$ (5.1) donnée en % (Table 5.1.)).

La figure 5.1. (resp. 5.2. et 5.3.) donne une représentation graphique pour la solution analytique $P(\mathcal{X}, t)$ () et $P(MSA, t)$ (resp. $P(MSM, t)$ et $P(MT, t)$) (annexe 3) dans son barycentre de taux de substitution (Table 5.4.). La courbe $P(\mathcal{X}MSA, t)$ (resp. $P(MSM, t)$, $P(MT, t)$) croise $P(\mathcal{X}, t)$ à $t_c \approx 2.51$ (resp. 2.29 et 2.69) et est corrélée avec les gènes réels des génomes d'archaea MSA (resp. MSM, MT) à $t_a \approx 3.24$ (resp. 3.44, 5.48) car $\Pr(MSA, \mathcal{X}, 3.24) \approx \Pr(MSA, \mathcal{X}, G_{MSA}) = 1.22\%$ (resp. $\Pr(MSM, \mathcal{X}, 3.44) \approx \Pr(MSM, \mathcal{X}, G_{MSM}) = 2.54\%$, $\Pr(MT, \mathcal{X}, 5.48) \approx \Pr(MT, \mathcal{X}, G_{MT}) = 2.70\%$).

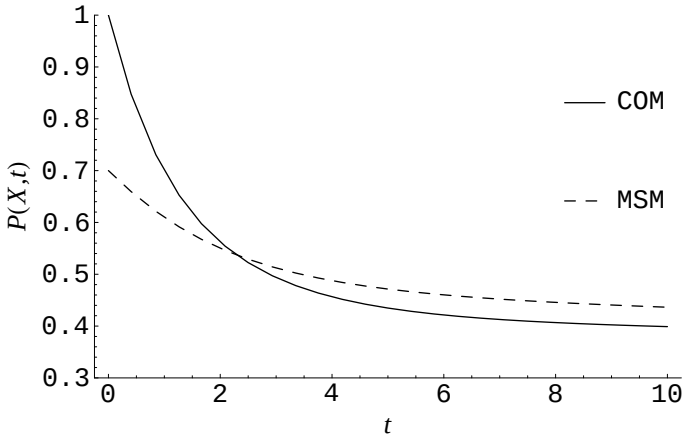

FIG. 5.2 – Evolution du code circulaire commun $\mathcal{X}$ et du code circulaire d'archaea MSM dans son barycentre des taux de substitution (en %) : $a = 15.0$, $b = 1.4$, $c = 19.0$, $d = 20.3$, $e = 18.3$ et $f = 26.0$ (Table 5.4.). La courbe $P(MSM, t)$ croise $P(\mathcal{X}, t)$ à $t_c \approx 2.29$ et est corrélée aux gènes réels du génome d'archaea MSM à $t_a \approx 3.44$.

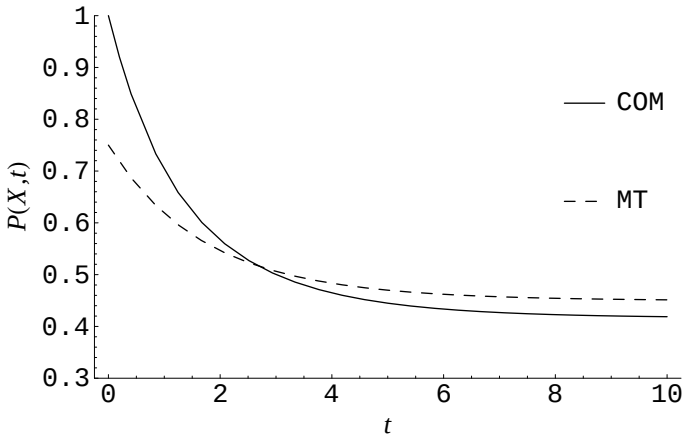

FIG. 5.3 – Evolution du code circulaire commun $\mathcal{X}$ et du code circulaire d'archaea MT dans son barycentre de taux de substitution (en %) : $a = 17.9$, $b = 0.02$, $c = 17.2$, $d = 21.9$, $e = 16.3$ et $f = 26.7$ (Table 5.4.). La courbe $P(MT, t)$ croise $P(\mathcal{X}, t)$ à $t_c \approx 2.69$ et est corrélée aux gènes réels du génome d'archaea MT à $t_a \approx 5.48$.

5.5 Discussion du modèle d'évolution

Un nouveau modèle d'évolution analytique a été développé afin de généraliser plusieurs modèles précédents ([JUK69], [KIM80], [ARQ98]). Il a été appliqué pour dériver les probabilités d'évolution du code circulaire commun $\mathcal{X}$ et des 15 codes d'archaea X en fonction du temps t et de 6 paramètres de substitutions associés aux transitions et aux transversions sur les 3 sites de trinucléotides.

De façon surprenante, tous les codes d'archaea X peuvent dériver du code commun $\mathcal{X}$ par des mutations aléatoires pour certaines valeurs des 6 paramètres de substitution. En effet, le modèle démontre qu'il est possible de trouver une différence de probabilité positive $\Pr(X, \mathcal{X}, t)$ (5.10) pour tous les codes d'archaea (Table 5.3.). De plus, une forte corrélation a été mise en évidence pour 3 codes d'archaea MSA, MSM et MT. Ainsi, la différence de probabilité $\Pr(X, \mathcal{X}, t)$ (5.11) obtenue pour ce modèle peut être supérieure à la différence de probabilité $\Pr(X, \mathcal{X}, G)$ observée dans les génomes (Table 5.4. et Figures 5.1., 5.2., 5.3.). Les temps d'intersection t_i sont 2.51, 2.29 et 2.69 pour MSA, MSM et MT respectivement, et le temps t_a correspondant aux probabilités actuelles 3.24, 3.44 et 5.48 respectivement. Les valeurs t_a suggèrent un temps d'évolution croissant de MSA, MSM à MT. On remarque que le temps d'intersection le plus court n'implique pas nécessairement le temps actuel le plus court : $t_c = 2.29$ pour MSM et $t_a = 3.24$ pour MSA. Une forte corrélation avec les 12 autres codes d'archaea nécessiterait une amélioration du modèle, en ajoutant d'autres paramètres dans le cadre d'un modèle numérique d'évolution ou en considérant une matrice de mutation non-symétrique.

Les valeurs faibles des barycentre du taux de substitutions permettent de proposer une classification de l'évolution des 15 codes d'archaea en 3 classes principales : une classe contenant le code TPA, une classe contenant les deux 2 codes HB et MP et une classe contenant les 12 codes restants qui subdivisent en 5 classes (Table 5.3).

Le code TPA est l'unique code d'archaea avec des faibles taux de substitution sur le $3^{ème}$ site des trinucléotides $r < 10\%$ (Table 5.3). Cette situation est en contradiction avec la dégénérescence du code génétique [Erm01]. Ce résultat suggère que le code TPA est le seul parmi les 15 codes d'archaea qui n'aurait pas évolué à partir du code circulaire commun $\mathcal{X}$. De nombreux gènes de l'archaea TPA ont été acquis par transfert latéral (acquisition par un organisme de matériel génétique d'une autre espèce qui sera conservé au cours de l'évolution) à partir de l'archaea SLS qui est le seul autre archaea à vivre dans

un environnement thermoacidophilique [Rue00].

Les codes HB et MP ont des taux faibles de substitution dans le $2^{ème}$ site des trinucléotides, i.e. $q < 20\%$ (Table 5.3). Le fait que HB et MP appartiennent à la même classe d'évolution peut être expliqué biologiquement par certaines de leur similarité, en particulier leur forte salinité intracellulaire qui implique certains gènes spécifiques qui n'apparaissent pas dans d'autres archaea [Sle02].
Les 12 autres codes d'archaea sont classés en 5 classes $\mathcal{C}_{b_1}$, $\mathcal{C}_{b_2}$, $\mathcal{C}_{b_3}$, $\mathcal{C}_{b_4}$ et $\mathcal{C}_{b_5}$ en fonction d'un taux faible de transversion dans le 1^{er} site des trinucléotides (b) (Table 5.3). Dans ces 5 classes de codes, le taux r est plus élevé que q qui est lui-même plus élevé que p ($r > q > p$ dans $\mathcal{C}_b$, Table 5.3), en accord avec la dégénérescence du code génétique où le taux de substitution est le plus fréquent dans le $3^{ème}$ site [Erm01]. De plus, comme les transversions sont associées à 2 mutations possibles et les transitions à une seule mutation (voir les matrices A, B et C de la section 5.3), les transitions sont plus fréquentes que les transversions pour chacun des 3 sites des 5 classes de codes ($a > b/2$, $c > d/2$ et $e > f/2$ dans $\mathcal{C}_b$ à l'exception d'un cas parmi les 36, PB dans le $3^{ème}$ site, Table 5.3). Cette observation est en accord avec les propriétés chimiques des nucléotides (un cycle carbone-nitrogène pour les pyrimidines et deux cycles carbone-nitrogène pour les purines) et avec la complémentarité des bases qui montrent un biais dans le taux de transition/transversion des génomes ([Och03], [Ros03]).

Les variations des courbes $P(\mathcal{X}, t)$ du code circulaire commun $\mathcal{X}$ et $P(X, t)$ des 15 codes circulaires d'archaea X donnant les probabilités d'occurrence de leurs trinucléotides en fonction de 6 paramètres de substitutions soumis à un processus d'évolution aléatoire, ne peuvent être prévues sans modélisation. En effet, certaines solutions analytique sont la somme de plusieurs termes exponentiels (23 termes pour le numérateur $P(\mathcal{X}, t)$ ()). Les différences de probabilités entre les trinucléotides des gènes primitifs (au temps $t = 0$) ont toujours une influence même après un grand nombre de substitutions dans les gènes (par exemple, au $t = 10$ dans les figures 1, 2 et 3). En effet, les traces des gènes primitifs générées par ces variations de trinucléotides, peuvent toujours être observées, même après une longue période de mutations aléatoires.

L'interprétation biologique de ce modèle d'évolution conduit à supposer que les gènes primitifs (à $t = 0$), sont construits à partir de trinucléotides du code circulaire commun $\mathcal{X}$. Seul 20 parmi les 64 trinucléotides ont été nécessaires. Les 20 types de trinucléotides ainsi que leur type de concaténation sont déterminés dans ce modèle. Ainsi, les 20 trinucléotides

sont définis par l'ensemble $\mathcal{X}$ qui est un code C^3 maximal et auto-complémentaire (Section 3.8). De plus, la concaténation indépendante et équiprobable de ces 20 trinucléotides est le plus simple type de concaténation possible et donc compatible avec les premier stades de l'évolution des gènes. Une concaténation Markovienne des trinucléotides (basée sur une matrice stochastique) aurait été trop complexe dans les temps primitifs.

La méthode développée peut être appliquée à d'autres problèmes d'évolution. En particulier, les valeurs propres obtenues peuvent être utilisés pour développer des modèles d'évolution similaires basés sur des matrices de mutations de trinucléotides à 6 paramètres. La matrice de mutation de trinucléotides peut également être utilisée dans des algorithmes comme la reconstruction d'arbre phylogénétique et l'alignement de séquences.

Chapitre 6

conclusions et perspectives

De nouveaux codes circulaires ont été identifiés dans les génomes de procaryotes. Le code $\mathcal{X}$ n'a pas été retrouvé pour les organismes. La méthode FPTF appliquée à l'ensemble des séquences permet cependant de retrouver un code très proche de $\mathcal{X}$: seul le codon GGT est modifié, remplacé par GTG. Le signal caractéristique de $\mathcal{X}$ est donc présent lors d'une analyse sur une population moyenne mais d'autres ensembles de codons préférentiels sont identifiés lors d'une étude par organisme.

Une étude plus générale a été menée pour tenter de comprendre la contribution des gènes d'eucaryotes ou de procaryotes à l'identification du code circulaire $\mathcal{X}$ dans les populations de séquences codantes suffisamment grandes. Au sein de notre équipe, une base de données, regroupant les gènes suivant des critères phylogénétique et fonctionnelles a été développée (projet GOTA pour Gene Ontologie and TAxonomie, réalisé avec M. Ahmed). Les séquences peuvent être classées par taxons ce qui généralise l'étude de génomes. La totalité des séquences d'une espèce peut par exemple être étudiée. Elle peut être combinée à l'étude des séquences caractérisées par un même terme de l'ontologie GO (Gene Ontologie). Une ontologie est une classification hiérarchisée des connaissances. GO est l'ontologie biologique la plus couramment utilisée. Elle nous permet de regrouper les gènes par fonction. Toutes les séquences de la base de données nucléiques EMBL adaptées à notre étude (fonction renseignée séquences non partielles, ...) sont copiées dans GOTA et classées. Une application permet d'appliquer la méthode FPTF aux populations de gènes générées par GOTA. L'étude est encore récente et les résultats obtenus sont en cours d'analyse. Elle devrait permettre d'e déterminer les ensembles de gènes qui contribue à la présence du signal caractéristique de $\mathcal{X}$ dans les populations importantes de séquences codantes de procaryotes et d'eucaryotes.

Les codes circulaires identifiés ont plusieurs propriétés intéressantes. En particulier, ils sont synchronisants. Pour un mot formé par la concaténation de mots d'un de ces codes circulaires, la lecture de courtes suites nucléotides, quelque soit la position, permet de retrouver la phase de construction. Par conséquent, les processus biologiques ne nécessiteraient pas de partir du site d'initiation pour agir sur la séquence en phase. Pour les codes identifiés, le délai de synchronisation est au maximum de l3 nucléotides. Cependant, les longs mots ambigus entre deux phases sont rares. Dans la majorité des cas, le nombre de lettres à lire pour retrouver la phase est à peu près de 2 trinucléotides. Même pour les séquences qui ne sont pas formées uniquement de mots de codes circulaires, les ensembles de codon préférentiels peuvent aider à retrouver la phase de lecture (voir 4.5). Les mots

des codes circulaires peuvent être aussi envisagés comme marqueur. Les ensembles $\mathcal{C}^3$ permettent signaliser les 3 phases. Les codes circulaires génèrent un grand nombre de mots différents, qui peuvent généralement s'adapter aux séquences en respectant la contrainte sur les acides aminés indispensables pour que la protéine synthétisée soit fonctionnelle.

Une fonction actuelle des codes circulaires dans les gènes n'a pas encore pu être identifiée. Plusieurs pistes sont actuellement explorées. Le maintien de la phase de lecture correcte par le ribosome est un mécanisme non encore totalement expliqué. Des motifs dans l'ARNm peuvent provoquer une perte de phase (voir 1.5). L'étude du signal associé à $\mathcal{X}$ dans les séquences provoquant un décalage dans la lecture est un sujet de recherche intéressant.

Bibliographie

[AE98] Akashi, H., Eyre-Walker, A., (1998). *Translational selection and molecular evolution.* Curr. Opin. Genet. Dev. 8, 688-693.

[AFM98] Arquès, D.G., Fallot, J.-P., Michel, C.J., (1998). *An evolutionary analytical model of a complementary circular code simulating the protein coding genes, the 5' and 3' regions.* Bull. Math. Biol. 60, 163-194

[AM95] Arquès, D.G., Michel C.J., (1996). *A Possible code in the genetic code.* STACS 95, 640-651.

[AM96] Arquès, D.G., Michel C.J., (1996). *A complementary circular code in the protein coding genes.* J. Theor. Biol. 182, 45-58.

[AM97] Arquès, D.G., Michel C.J., (1997). *A circular code in the protein coding genes of mitochondria.* J. Theor. Biol. 189, 273-290.

[Bar03] Baranov, P. V., Gurvich O. L., Hammer, A. W. , Gesteland, R. F. and Atkins, J. F. (2003).*RECODE 2003.* Nucleic Acids Res. 31, 8789.

[BBD79] Barrell, B. G., Bankier, A. T., Drouin, J. A. (1979). *Different genetic code in human mitochondria.* Nature 282, 189-194.

[Bas99] Bassino F. (1999). *Generating Fonctions of circular codes.* Advances in Applied Mathematics , 22 : 1, p. 1-24, 1999.

[Bea93] Béal M.P. (1993). *Codage symbolique.* Masson.

[Ber00] Bernander, R., (2000). *Chromosome replication, nucleotid segregation and celldivision in archaea.* Trends in Microbiol. 8, 278-283.

[Bla97] Blattner, F. R., III, G. P., Bloch, C. A., Perna, N. T., Burland, V. et al (1997). *The complete genome sequence of Escherichia coli K-12.* Science 277, 1453-1461.

[BP85] Berstel J. and Perrin D. (1985). *Theory of codes.* Academic Press.

[Bre57] Brenner, S. (1957). *On the impossibility of all overlapping triplet codes in information transfer from nucleic acid to proteins.* Proceedings of the National Academy of Sciences of the U.S.A. 43, 687-694.

[Bru91] Bruyere V. (1991). *Maximal codes with bounded deciphering delay.* Theoret. Comput. Sci., 84 53-76

[BS97] Berg, O.G, Silva, P.J.N., (1997). *Codon bias in Escherichia coli : the influence of codon context on mutation and selection.* Nucleic Acids Res. 25, 1397-1404.

[Bul96] Bult, C. J., White, O., Olsen, G. J., Zhou, L., Fleischmann, R. D. et al (1996). *Complete genome sequence of the methanogenic Archaeon, Methanococcus jannaschii.* Science 273, 1058-1072

[CG57] Crick, F. H. C., J. S. Griffith and L. E. Orgel. (1957). *Codes without commas.* Proceedings of the National Academy of Sciences of the U.S.A. 43, 416-421.

[Cla01] Clark, M. A., Baumann, L., Thao, M. L., Moran, N. A., Baumann, P. (2001). *Degenerative Minimalism in the Genome of a Psyllid Endosymbiont.* J. Bacter 183 (6) 1853-1861.

[CMK99] Campbell, A., Mrázek, J., Karlin, S., (1999). *Genomic signature comparisons among prokaryote, plasmid, and mitochondrial DNA.* Proc. Natl. Acad. Sci. U.S.A. 96, 9184-9189.

[Cop03] Copeland P. R. (2003). *Regulation of gene expression by stop codon recoding : selenocysteine.* Gene, 312, 17-25.

[Cri66] Crick. F. (1966). *Codon-anticodon pairing : the wobble hypothesis.* J Mol Biol, 1966 Aug ;19(2), 548-55.

[Cri68] Crick, F. (1968). *The origin of the genetic code.* J. Mol. Biol. 38, 367-379.

[CRM05] Cobucci-Ponzano, B., Rossi M. and Moracci M. (2005). *Recoding in Archaea.* Molecular Microbiology 55, 339-348.

[Duj96] Dujon, B (1996). *The yeast genome project : what did we learn ?* Trends Genet 12, 263-270.

[ER85] Ehrenfeucht A. , Rozenberg G., (1985). *Each regular code is included in a regular maximal code.* RAIRO Inform Theor. Appl., 20, 89-96,

[Erm01] Ermolaeva, M.D., (2001). *Synonymous codon usage in bacteria.* Curr Issues Mol Biol Oct 3, 91-97.

[ES78] Eigen M., Schuster P. (1978). *The hypercycle. A principle of natural self-organisation. Part C : the realistic hypercycle.* Naturwissenchaften 65, 341-36

[Far96] Farabaugh, P. J. (1996). *Programmed translational frameshifting.* Microbiol Rev 60(1), 103-134.

[Fre00] Freeland, S., Knight, R.D., Landweber, L.F. and Hurst, L.D. (2000). *Early fixation of an optimal genetic code*. Molecular Biology and Evolution 17, 511-518.

[FH98] Freeland, J., Hurst. L.D. (1998). *The Genetic Code Is One in a Million*, J. Mol. Evol. 47, 238-248.

[Fle95] Fleischmann, R. D., Adams, M. D., White, O., Clayton, R. A., Kirkness, E. F. et al (1995). *Whole genome random sequencing and assembly of aemophilus influenzae Rd.* Science 269, 496-512.

[FM03] Frey, G., Michel, C.J., (2003). *Circular codes in archaeal genomes*. J. Theor. Biol. 223, 413-431

[For01] Forterre, P., (2001). *Genomics and early cellular evolution. The origin of the DNA world*. C. R. Acad. Sci. III 324, 1067-1076.

[Gam54a] Gamow, G. (1954a). *Possible relation between deoxyribonucleic acid and protein structures*. Nature 173, 318.

[Gam54b] Gamow, G. (1954b). *Possible mathematical relation between deoxyribonucleic acid and proteins*. Det Klongelige Danske Videnskabernes Selskab, Biologiske Meddelser 22, 1-13.

[Gam56] Gamow, George, Alexander Rich and Martynas Ycas. (1956). *The problem of information transfer from nucleic acids to proteins*. Advances in Biological and Medical Physics, Vol.4.,pp.23-68.New-York Academic Press.

[GG65] Golomb S.W. , B. Gordon. (1965). *Codes with bounded synchronization delay*. Inform. Control 8, 355-372

[GGG80] Grantham, R., Gautier, C., Gouy, M., Mercier, R., Pave, A., (1980). *Codon catalog usage and the genome hypothesis*. Nucleic Acids Res. 8, 49-62.

[GGW58] Golomb S.W., B Gordon, L.R. Welch. (1958). *Comma-free codes*. Canad. J. Math. 10, 202-209

[GM59] Gilbert E. N., Moore E. F. (1965). *Variable Length Binary Encodings*. Bell System Tech. J., 38 933-967

[Gue00] Guesnet Y. (2000).*On codes with finite interpreting delay : A defect theorem*. Theoret. Informatics Appl., 47-59

[Gue01] Guesnet Y. (2001).*Codes et interprétations*. Thèse de doctorat, Université de Rouen.

[GWA92] Gesteland, R. F., Weiss R. B. and Atkins J. F. (1992). *Recoding : Reprogrammed genetic decoding*. Science 257, 1640-1643.

[GWD58] Golomb S.W., L.R. Welch, M. Delbrück. (1958). *Construction and properties of comma-free codes.* Biol. Medd. Dan. Vid. Selsk. 23(9).

[Hay98] Hayes, B. (1998). *The invention of the genetic code.* American Scientist 86(1) 8-14.

[HH92] Hogeweg, P., Hesper, B. (1992). *Evolutionary dynamics and the coding structure of sequences : Multiple coding as a consequence of crossover and high mutation rates.* Computers Chem 4, 300-314.

[Hol68] Holliday, R. (1968). *Genetic recombination in fungi.* Replication and recombination of genetic material. Australian Academy of Science,

[HKH93] Huynen, M. A. , Konings, D. A., Hogeweg, P. (1993). *Multiple coding and the evolutionary properties of RNA secondary structure.* J. Theor. Biol. 165, 251-267.

[Huf52] David A. Huffman. (1952). *A Method for the Construction of Minimum Redundancy Codes.* Proceedings of the Institute of Radio Engineers. 40 Number 9, 1098-1101.

[Hun98] Hung, M., Patel, P. , Davis, S. and Green S. R. (1998). *Importance of ribosomal frameshifting for human immunodeficiency virus type 1 particle assembly and replication.* J Virol 72(6), 4819-4824.

[Ike85] Ikemura, T., (1985). *Codon usage and tRNA content in unicellular and multicellular organisms.* Mol. Biol. Evol. 2, 12-34.

[Ino00] Inokuchi, Y., Hirashima, A., Sekine, Y. Janosi, L., Kaji, A. (2000). *Role of ribosome recycling factor (RRF) in translational coupling.* EMBO Journ. 19(14) 3788-3798.

[IS04] Ibba, M., and Söll, D. (2004). *Aminoacyl-tRNAs : setting the limits of the genetic code.* Genes Dev 18, 731-738.

[JB86] Jukes, T.H., Bhushan, V., (1986). *Silent nucleotide substitutions and G+C content of some mitochondrial and bacterial genes.* J. Mol. Evol. 24, 39-44.

[JC69] Jukes, T.H., Cantor, C.R., (1969). *Evolution of protein molecules.* Academic Press, New York, pp. 21-132.

[Jig63] Jiggs. B.H. (1963). *Recent results in comma-free codes.* Theoret. Comput. Math. 15, 178-187.

[Jor93] Jorgensen, F., F. M. Adamski, W. P. Tate and Kurland C. G. (1993). *Release factor-dependent false stops are infrequent in Escherichia coli.* J Mol Biol 230(1), 41-50.

[Juk96] Jukes. T.H. (1996). *On the Prevalence of Certain Codons ("RNY") in Genes for Proteins.* J. Mol. 42, 377-381.

[Kim80] Kimura, M., (1980). *A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences.* J. of Mol. Evolu. 16, 111-120.

[KL02] Konu, O., Li, M.D., (2002). *Correlations between mRNA expression levels and GC contents of coding and untranslated regions of genes in rodents.* J. Mol. Evol. 54, 35-41.

[KL97] Koch A. J., Lehmann J. (1997).*About a symmetry in genetic code.* J. Theor. Biol., 189 171-174

[KFL01] Knight R.D., Freeland S.J., Landweber L.F. (2001). it Rewiring the keyboard : evolvability of the genetic code. Nat Rev Genet. 2, 49-58.

[Kon92] Konings, D. A. (1992). *Coexistence of multiple codes in messenger RNA molecules.* Comput. Chem. 16, 153-163.

[Kra00] Krakauer, D. C. (2000). *Stability and evolution of overlapping genes.* Int J Org Evol 54, 731-739.

[KSH95] Konecny J., Schöniger M. and Hofacker G. L. (1995). *Complementary coding conforms to the primeval comma-less coding.* J. Theor. Biol. 173, 263-270.

[Kur92] Kurland, C. G. (1992). *Translational accuracy and the fitness of bacteria.* Annu Rev Genet 26(29), 29-50.

[Lam03] Lam. N.H. (2003). *Completing comma-free codes.* Theorical Computer Science 301, 399-415

[Lan05] Lange, K., (2005). *Applied Probability.* Springer-Verlag, New York.

[Lec85] Leconte M. (1985). *Codes sans répétition.* Thèse, Université Paris VII.

[LM01] Lacan, J., Michel. C.J. (2001). *Analysis of a circular code model.* J. Theor. Biol. 213, 159-170

[Mar67] Markov A. (1967). *An example of an independant system of word which cannot be included in a finite complete system.* Mat. Zametki, 1 87-90

[MR83] Metropolis N, Rota G. C. (1983). *Witt vectors and the algebra of necklaces.* Adv. Math., 50, 95-125,

[Nam04] Namy, O., Rousset J.P., Napthine S. and Brierley I. (2004). *Reprogrammed Genetic Decoding in Cellular Gene Expression.* Molecular Cell, Vol. 13, 157-168.

[Niv66] Nivat M. (1966). *Elements de la théoride générale des codes.* Automata Theory, 278-294, Academic Press, New York

[NMM61] Nirenberg, Marshall W., and J. Heinrich Matthaei. (1961). *The dependence of cell-free protein synthesis in E. coli upon naturally occuring or synthetc polyribonucleotides.* Proceedings of the National Academy of Sciences of the U.S.A. 47, 1588-1602.

[NS01] Neraud J., Selmi C. (2002). *Locally complete sets and finite decomposable codes.* Theoret. Comput. Sci., 273 185-196

[Och03] Ochman, H., (2003). *Neutral mutations and neutral substitutions in bacterial genomes.* Mol. Biol. Evol. 20, 2091-2096.

[Par89] Parker, J. (1989). *Errors and alternatives in reading the universal genetic code.* Microbiol Rev 53(3), 273-298.

[Pav97] Pavesi, A., De Iaco, B., Granero, M.I., Porati, A. (1997). *On the informational content of overlapping genes in prokaryotic and eukaryotic viruses.* J Mol Evol 44, 625-631.

[Pir04] Pirillo G et Pirillo M. A. (2005). *Growth function of self complementary circular codes.* Biology Forum, 97 97-110

[Pla98] Plat, T. (1998). *RNA structure in transcription elongation, termination and antitermination.* RNA structure and function, ed. M. Grunberg-Manago. pp. 541-574 Cold Spring Harbour Laboratory Press.

[Plu94] Pluhar, W. (1994). *The molecular basis of wobbling - an alterative hypothesis.* Journal of Theoretical Biology 169(3), 305-312.

[RDV99] Rocha, E. P. C., Danchin, A., Viari, A. (1999). *Translation in Bacillus subtilis : roles and trends of initiation and termination, insights from a genome analysis.* Nucl. Acids res. 27, 3567-3576.

[RSK03] Rosenberg, M.S., Subramanian, S., Kumar, S., (2003). *Patterns of transitional mutation biases within and among mammalian genomes.* Mol. Biol. Evol. 20, 988-993.

[Rue00] Ruepp, A., et al., (2000). *The genome sequence of the thermoacidophilic scavenger thermoplasma acidophilum.* Nature 407, 508-513.

[San78] Sanger, F., Coulson, A. R., Friedman, T., Air, G. M., Barrel, B. G. et al (1978). *The nucleotide sequence of bacteriophage phiX174.* J. Mol. Biol. 125, 225-246.

[Sch56] Schützenberger M. P. (1956). *Une théorie algébrique du codage.* Séminaire Dubreil-Pisot, Institut H. Poincarré, Exposé 15

[SD74] Shine J. and Dalgarno L. (1974). *The 3'-terminal sequence of Escherichia coli 16S ribosomal RNA : complementarity to nonsense triplets and ribosome binding sites.* Proc. Natl. Acad. Sci. USA 71, 1342-1346.

[She81a] Shepherd J.C.W. (1981a). *Method to determine the reading frame of a protein from the purine/pyrimidine genome sequence and its possible evolutionary justification.* Proceedings of the National Academy of Sciences of the U.S.A. 78, 1596-1600

[She81b] Shepherd J.C.W. (1981b). *Periodic correlations in DNA sequences and evidence suggesting their evolutionary origin in a comma-less genetic code.* J. Mol. Evol. 17, 94-102.

[Shp86] Shpaer, E.G., (1986). *Constraints on codon context in Escherichia coli genes. Their possible role in modulating the efficiency of translation.* J. Mol. Biol. 188, 555-564.

[Sle02] Slesarev, A., et al., (2002). *The complete genome of hypermetrohile Methanopyrus kandleri AV19 and monophyly of archaeal methanogens.* Proc. Natl. Acad. Sci. U.S.A. 99, 4644-4649.

[Sha05] Sharp PM, Bailes E, Grocock RJ, Peden JF, Sockett RE (2005). *Variation in the strength of selected codon usage bias among bacteria.*. Nucleic Acids Res. 33(4), 1141-1153.

[SP53] Sardinas A. A., Patterson C. (1953). *A necessary and sufficient condition for the unique decomposition of coded message.* IRE Internat. Conv. Rec., 8 104-108

[SS86] Shapiro M.B. , Senapathy P. (1986). *RNA splice junctions of different classes of eukaryotes : sequence statistics and functional implications in gene expression.* Nucl. Acids Res. 15, 7155-7174.

[Sha99] Shapiro, R. (1999). *Prebiotic cytosine synthesis : A critical analysis and implications for the origin of life.* Proc. Natl. Acad. Sci. U.S.A. 96, 4396-401.

[Sta01] Stahl, G., Ben Salem, S., Li, Z., McCarty, G., Raman, A., Shah, M. and Farabaugh P. J. (2001). *Programmed translational frameshifting in the yeast Saccharomyces cerevisiae results from disruption of translational error correction.* Cold Spring Harbor Symp. Quant. Biol. 66, 249-258.

[Tri87] Trifonov, E.N. (1987). *Translation framing code and frame-monitoring mechanism as suggested by the analysis of mRNA and 16S rRNA nucleotide sequences.* J. Mol. Biol. 194, 643-652

[Tri89] Trifonov, E.N. (1989). *The multiple codes of nucleotide sequences.* Bull Math Biol 51, 417-432.

[VR92] Vellanoweth, R. L., Rabinowitz, J. C. (1992). *The influence of ribosome-binding-site elements on translational efficiency in Bacillus subtilis and Escherichia coli in vivo.* Mol. Microbiol. 6, 1105-1114.

[WC53] Watson, J.D., Crick, F.H.C. (1953). *A structure for deoxyribose nucleic acid.* Nature 171, 737-8.

[Woe65] Woese, C.R. (1965). *On the evolution of the genetic code.* Proc Natl Acad Sci U S A. 54(6), 1546-52.

[Woe77] Woese, C.R., Fox, G.E. (1977). *Phylogenetic structure of the prokaryotic domain : the primary kingdoms.* roc. Natl. Acad. Sci. U.S.A 1977 74, 5088-90

[Woe00] Woese, C.R., (2000). *Interpreting the universal phylogenetic tree.* Proc. Natl. Acad. Sci. U.S.A. 97, 8392-8396.

[WS99] Wagner, A., Stadler, P.F. (1999). *Viral RNA and evolved mutational robustness.* J. Exp. Zool. 285, 119-127.

Annexe A

Bibliographie personnelle

Articles en revues internationales avec comité de rédaction

[1] G. Frey and C. Michel, Circular codes in archaeal genomes, *Journal of Theoretical Biology*, 223, 413-431, 2003

[2] G. Frey and C. Michel, A Stochastic Evolutionary Model of Archaeal Circular Codes, *Journal of Chemistry and Computer Biology*, accepté, 2005

[3] G. Frey and C. Michel, Identification of circular codes in bacterial genomes and their use in a factorization method for retrieving reading frame of genes, *Journal of Chemistry and Computer Biology*, accepté, 2005

Communication dans un congrès international avec comité de sélection

[4] G. Frey and C. Michel, Circular codes in prokaryotic genomes, 3rd Workshops on Algorithms in Bioinformatics, Budapest, Hongrie, 2003

Communication dans des groupes nationaux, séminaires ou forum

[5] G. Frey, Codes dans les gènes de bactéries, Journées de l'indexation et de l'algorithme du texte, Lilles, décembre 2004

Annexe B

Abbréviations des archaea

Genome G_{AG} : Archeoglobus fulgidus (C_{72})

Genome G_{AP} : Aeropyrum pernix (C_{72})

Genome G_{HB} : Halobacterium sp.NCR-1 (C_0)

Genome G_{MC} : Methanoccocus jannashii (C_{73})

Genome G_{MP} : Methanopyrus kandleri (C_{74})

Genome G_{MSA} : Methanosarcina acetivorans (C_{75})

Genome G_{MSM} : Methanosarcina mazei (C_{76})

Genome G_{MT} : Methanothermobacter thermautotrophicus (C_{77})

Genome G_{PB} : Pyrobaculum aerophilum (C_{78})

Genome G_{PCA} : Pyroccocus abyssi (C_{79})

Genome G_{PCF} : Pyroccocus furiosus (C_{80})

Genome G_{PCH} : Pyroccocus horikoshii (C_{81})

Genome G_{SLS} : Sulfolobus solfataricus (C_{82})

Genome G_{SLT} : Sulfolobus tokodaii (C_{83})

Genome G_{TPA} : Thermoplasma acidophilium (C_{84})

Genome G_{TPV} : Thermoplasma volcanium (C_{85})

Annexe C

Formules analytiques d'évolution

Dénominateur D

$$D = 150 - e^{\lambda_4 t} + 4e^{\lambda_6 t} - e^{\lambda_7 t} + e^{\lambda_{21} t} + 3e^{\lambda_{22} t} + 2e^{\lambda_{23} t} - 2e^{\lambda_{24} t} + e^{\lambda_{25} t} + 3e^{\lambda_{26} t}. \quad \text{(C.1)}$$

Avec le dénominateur D et les valeurs d'Eigen λ_k de M (voir 5.3), les formules analytique d'évolution $P(X,t)$ obtenues pour les 15 codes circulaires d'archaeas X sont

$$P(AG/AP,t) = \frac{1}{D}\Big(50 + 5e^{\lambda_2 t} + 15e^{\lambda_5 t} - e^{\lambda_7 t} + 11e^{\lambda_9 t} + 18e^{\lambda_{11} t} - 2e^{\lambda_{13} t} + e^{\lambda_{15} t} + 2e^{\lambda_{18} t} - e^{\lambda_{19} t} + 2e^{\lambda_{21} t} + 9e^{\lambda_{22} t} + e^{\lambda_{23} t} - e^{\lambda_{26} t} + 3e^{\lambda_{27} t}\Big)$$

$$P(HB,t) = \frac{1}{2D}\Big(100 + 10e^{\lambda_2 t} + 10e^{\lambda_5 t} + 8e^{\lambda_6 t} + 24e^{\lambda_9 t} + 6e^{\lambda_{10} t} + 42e^{\lambda_{11} t} - 2e^{\lambda_{12} t} - 2e^{\lambda_{13} t} - e^{\lambda_{17} t} + 3e^{\lambda_{18} t} + e^{\lambda_{19} t} + e^{\lambda_{20} t} + 4e^{\lambda_{21} t} + 8e^{\lambda_{22} t} + 12e^{\lambda_{26} t}\Big)$$

$$P(MC,t) = \frac{1}{2D}\Big(100 + 5e^{\lambda_2 t} + 30e^{\lambda_5 t} + 4e^{\lambda_6 t} - e^{\lambda_7 t} - 11e^{\lambda_9 t} - 6e^{\lambda_{11} t} + 3e^{\lambda_{12} t} + 6e^{\lambda_{13} t} - e^{\lambda_{14} t} + e^{\lambda_{15} t} + 20e^{\lambda_{17} t} - e^{\lambda_{18} t} + 3e^{\lambda_{19} t} + e^{\lambda_{21} t} + 13e^{\lambda_{22} t} + 3e^{\lambda_{23} t} - 5e^{\lambda_{24} t} - 3e^{\lambda_{25} t} - 3e^{\lambda_{26} t} + 2e^{\lambda_{27} t}\Big)$$

$$P(MP,t) = \frac{1}{2D}\Big(100 + 20e^{\lambda_5 t} - 2e^{\lambda_7 t} + 23e^{\lambda_9 t} + 3e^{\lambda_{10} t} + 42e^{\lambda_{11} t} - 2e^{\lambda_{12} t} - e^{\lambda_{13} t} + e^{\lambda_{14} t} + 2e^{\lambda_{15} t} + 4e^{\lambda_{18} t} + 2e^{\lambda_{20} t} + 6e^{\lambda_{21} t} + 16e^{\lambda_{22} t} + 2e^{\lambda_{23} t} - 2e^{\lambda_{25} t} + 4e^{\lambda_{26} t} + 6e^{\lambda_{27} t}\Big)$$

$$P(MSA,t) = \frac{1}{2D}\Big(100 + 20e^{\lambda_2 t} + e^{\lambda_4 t} + 35e^{\lambda_5 t} + 4e^{\lambda_6 t} - 2e^{\lambda_7 t} + 9e^{\lambda_9 t} - 2e^{\lambda_{10} t} + 30e^{\lambda_{11} t} + e^{\lambda_{12} t} + 2e^{\lambda_{13} t} + e^{\lambda_{14} t} + 3e^{\lambda_{15} t} + 18e^{\lambda_{17} t} - e^{\lambda_{18} t} + 3e^{\lambda_{19} t} + 2e^{\lambda_{20} t} + e^{\lambda_{21} t} + 5e^{\lambda_{22} t} + e^{\lambda_{23} t} + e^{\lambda_{24} t} + e^{\lambda_{25} t} + 9e^{\lambda_{26} t} - 2e^{\lambda_{27} t}\Big)$$

$$P(MSM,t) = \frac{1}{D}\Big(50 + 10e^{\lambda_2 t} + e^{\lambda_4 t} + 20e^{\lambda_5 t} + 4e^{\lambda_6 t} - e^{\lambda_7 t} + 2e^{\lambda_9 t} - e^{\lambda_{10} t} + 12e^{\lambda_{11} t} + e^{\lambda_{13} t} + 2e^{\lambda_{15} t} + 8e^{\lambda_{17} t} - e^{\lambda_{18} t} + 2e^{\lambda_{19} t} + e^{\lambda_{20} t} + e^{\lambda_{25} t} + 3e^{\lambda_{26} t} - 2e^{\lambda_{27} t}\Big)$$

$$P(MT,t) = \frac{1}{2D}\Big(100 + 20e^{\lambda_2 t} + e^{\lambda_4 t} + 35e^{\lambda_5 t} + 4e^{\lambda_6 t} - 2e^{\lambda_7 t} + 15e^{\lambda_9 t} + 2e^{\lambda_{10} t} + 30e^{\lambda_{11} t} + e^{\lambda_{12} t} + e^{\lambda_{14} t} + 3e^{\lambda_{15} t} + 12e^{\lambda_{17} t} + e^{\lambda_{18} t} - e^{\lambda_{19} t} + 2e^{\lambda_{20} t} + 3e^{\lambda_{21} t} + 11e^{\lambda_{22} t} + e^{\lambda_{23} t} + e^{\lambda_{24} t} - e^{\lambda_{25} t} + 3e^{\lambda_{26} t} - 2e^{\lambda_{27} t}\Big)$$

$$P(PB,t) = \frac{1}{2D}\Big(100 + 5e^{\lambda_2 t} + 30e^{\lambda_5 t} - 4e^{\lambda_6 t} - 3e^{\lambda_7 t} + 18e^{\lambda_9 t} + e^{\lambda_{10} t} + 36e^{\lambda_{11} t} + e^{\lambda_{12} t} + e^{\lambda_{13} t} + 3e^{\lambda_{15} t} + 10e^{\lambda_{17} t} + 5e^{\lambda_{18} t} + e^{\lambda_{19} t} + 3e^{\lambda_{21} t} + 17e^{\lambda_{22} t} + 3e^{\lambda_{23} t} + e^{\lambda_{24} t} - e^{\lambda_{25} t} + 9e^{\lambda_{26} t} + 4e^{\lambda_{27} t}\Big)$$

$$P(PCA,t) = \frac{1}{2D}\Big(100 + 15e^{\lambda_2 t} + e^{\lambda_4 t} + 35e^{\lambda_5 t} - 3e^{\lambda_7 t} + 13e^{\lambda_9 t} + 3e^{\lambda_{10} t} + 30e^{\lambda_{11} t} + 2e^{\lambda_{12} t} - e^{\lambda_{13} t} + e^{\lambda_{14} t} + 4e^{\lambda_{15} t} + 12e^{\lambda_{17} t} + 2e^{\lambda_{18} t} - 4e^{\lambda_{19} t} + 2e^{\lambda_{20} t} + 4e^{\lambda_{21} t} + 12e^{\lambda_{22} t} - 2e^{\lambda_{25} t} - 2e^{\lambda_{26} t}\Big)$$

$$P(PCF,t) = \frac{1}{2D}\Big(100 + 5e^{\lambda_2 t} + 30e^{\lambda_5 t} - 4e^{\lambda_6 t} - 3e^{\lambda_7 t} + e^{\lambda_9 t} + 24e^{\lambda_{11} t} + 5e^{\lambda_{12} t} + 2e^{\lambda_{13} t} + e^{\lambda_{14} t} + 3e^{\lambda_{15} t} + 21e^{\lambda_{17} t} + 2e^{\lambda_{19} t} + e^{\lambda_{20} t} + e^{\lambda_{21} t} + 3e^{\lambda_{22} t} - 3e^{\lambda_{23} t} - e^{\lambda_{24} t} - e^{\lambda_{25} t} + e^{\lambda_{26} t} + 4e^{\lambda_{27} t}\Big)$$

$$P(PCH,t) = \frac{1}{2D}\Big(100 + 30e^{\lambda_5 t} - 2e^{\lambda_7 t} + 3e^{\lambda_9 t} + 5e^{\lambda_{10} t} + 24e^{\lambda_{11} t} + 4e^{\lambda_{12} t} + e^{\lambda_{13} t} - e^{\lambda_{14} t} + 4e^{\lambda_{15} t} + 21e^{\lambda_{17} t} + 3e^{\lambda_{18} t} + e^{\lambda_{19} t} - e^{\lambda_{20} t} + 8e^{\lambda_{22} t} - 6e^{\lambda_{23} t} - 2e^{\lambda_{24} t} - 2e^{\lambda_{25} t} + 2e^{\lambda_{26} t}\Big)$$

$$P(SLS,t) = \frac{1}{2D}\Big(100 + 5e^{\lambda_2 t} + 30e^{\lambda_5 t} + 4e^{\lambda_6 t} - e^{\lambda_7 t} - 10e^{\lambda_9 t} + e^{\lambda_{10} t} + 3e^{\lambda_{12} t} + 3e^{\lambda_{13} t} + 3e^{\lambda_{15} t} + 20e^{\lambda_{17} t} - e^{\lambda_{18} t} + 3e^{\lambda_{19} t} + e^{\lambda_{21} t} + 13e^{\lambda_{22} t} + e^{\lambda_{23} t} - 3e^{\lambda_{24} t} - 3e^{\lambda_{25} t} + 5e^{\lambda_{26} t} + 2e^{\lambda_{27} t}\Big)$$

$$P(SLT,t) = \frac{1}{D}\Big(50 + 5e^{\lambda_2 t} + 15e^{\lambda_5 t} + 4e^{\lambda_6 t} - 7e^{\lambda_9 t} - 2e^{\lambda_{10} t} - 6e^{\lambda_{11} t} + 3e^{\lambda_{13} t} + 10e^{\lambda_{17} t} - e^{\lambda_{18} t} + 3e^{\lambda_{19} t} + 6e^{\lambda_{22} t} + e^{\lambda_{23} t} - e^{\lambda_{24} t}\Big)$$

$$P\left(TPA,t\right)=\frac{1}{2D}\Big(100+15e^{\lambda_2 t}-e^{\lambda_4 t}+25e^{\lambda_5 t}-e^{\lambda_7 t}+24e^{\lambda_9 t}+2e^{\lambda_{10} t}+36e^{\lambda_{11} t}-2e^{\lambda_{12} t}-4e^{\lambda_{13} t}$$
$$+\ 2e^{\lambda_{17} t}+4e^{\lambda_{18} t}+2e^{\lambda_{21} t}+12e^{\lambda_{22} t}+2e^{\lambda_{23} t}+2e^{\lambda_{25} t}+4e^{\lambda_{26} t}+2e^{\lambda_{27} t}\Big)$$

$$P\left(TPV,t\right)=\frac{1}{2D}\Big(100+20e^{\lambda_2 t}+e^{\lambda_4 t}+35e^{\lambda_5 t}+12e^{\lambda_6 t}-2e^{\lambda_9 t}-e^{\lambda_{10} t}+18e^{\lambda_{11} t}+3e^{\lambda_{12} t}+3e^{\lambda_{13} t}$$
$$+\ 3e^{\lambda_{15} t}+19e^{\lambda_{17} t}-2e^{\lambda_{18} t}+6e^{\lambda_{19} t}+e^{\lambda_{20} t}+e^{\lambda_{21} t}-e^{\lambda_{22} t}-e^{\lambda_{23} t}+e^{\lambda_{24} t}-e^{\lambda_{25} t}+9e^{\lambda_{26} t}\Big)$$