Université de Montréal

Recherche de snoRNAs de type C/D dans le génome de *S.cerevisiae* en corrélation avec le signal de reconnaissance à l'enzyme RNT1P

par
Sébastien Christin

Département Bio-informatique
Faculté des études supérieures

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de Maître ès sciences (M.Sc.)
en Bio-informatique

11, 2006

Université de Montréal
Faculté des études supérieures

Ce mémoire intitulé:

Recherche de snoRNAs de type C/D dans le génome de *S.cerevisiae* en corrélation avec le signal de reconnaissance à l'enzyme RNT1P

présenté par:

Sébastien Christin

a été évalué par un jury composé des personnes suivantes:

Hervé Philippe,	président-rapporteur
Nadia El-Mabrouk,	directeur de recherche
Pascal Vincent,	membre du jury

Mémoire accepté le: ...27 mai 2007.....

RÉSUMÉ

Mots clés : ARN, snoRNA, RNT1P, *Saccharomyces cerevisiae*, algorithme de recherche, SVM, génomique comparative

Ce travail concerne la recherche de snoRNA de type boîte C/D dans le génome de *Saccharomyces cerevisiae*. La séquence transcrite présentant très peu de motifs conservés, il est très difficile de définir un consensus pour cette famille d'ARN. La recherche de motifs conservés pour trouver de nouveaux snoRNAs de type C/D n'est pas adéquate. En effet, avec les caractéristiques connues (boîtes C et D, hélice terminale), une recherche sur un génome relativement petit, *Saccharomyces cerevisiae*, retourne des milliers de possibilités. Si on ajoute le motif ASE, responsable de l'interaction avec l'ARN cible, on obtient un nombre raisonnable de candidats, mais on restreint la recherche aux snoRNAs responsables de la méthylation sur ce même ARN.

Un motif de clivage se trouve dans le voisinage de la séquence du snoRNA transcrite. Étant reconnu par une enzyme de type RNase III, la RNT1P, ce motif intervient dans la maturation du snoRNA. Une caractérisation et une fouille de ce motif dans le génome permettrait d'avoir un critère de sélection supplémentaire pour la découverte de snoRNAs, permettant de remplacer l'ASE et ainsi d'obtenir des candidats jusque-là ignorés.

Ayant à notre disposition des séquences de substrats à l'enzyme RNT1P, il nous a été possible de définir des caractéristiques générales permettant de circonscrire la recherche et d'avoir un nombre raisonnable de candidats snoRNAs. C'est un motif tige-boucle, avec une boucle à quatre de type DGNN, dont les trois premiers appariements sont de type Watson-Crick. Plusieurs méthodes ont été utilisées afin de caractériser davantage le motif tetraloop : statistique d'information mutuelle, recherche de motif simple, analyse de la structure d'hélice, classification par apprentissage machine (SVM).

Après avoir caractérisé le motif de reconnaissance à l'enzyme RNT1P, nous présentons une méthode de recherche des snoRNAs de type C/D en corrélation avec ce motif. Elle se base sur la recherche de séquences et d'hélices conservées, sur la validation des candidats par génomique comparative, et sur un algorithme de classification efficace.

Cette méthode nous a permis d'identifier 32 candidats snoRNAs potentiels. De ces 32 candidats, 3 présentent une différence de concentration de transcrit similaire à la famille des snoRNAs, faisant d'eux de bons candidats pour une validation expérimentale.

ABSTRACT

Keywords : RNA, snoRNA, RNT1P, *Saccharomyces cerevisiae*, searching algorithms, SVM, comparative genomic

This work presents a new method for the search of C/D box snoRNA in the *Saccharomyces Cerevisiae* genome. As the transcribed sequence has very few known conserved motifs (C box, D box, terminal stem), a search method based solely on the identification of such motifs gives rise to thousands of hits in a relatively small genome (*S.cerevisiae*). An additional constraint that has been considered in the literature to reduce this huge number of candidates is the use of another motif, ASE, which is required for the interaction with the target RNA. However, the search method is then restricted to the sub-family of C/D box snoRNAs that interacts with the given RNA.

Another characteristic of snoRNAs that has never been considered in a search method is the presence of a cleavage stem in the neighborhood of the snoRNA transcribed sequence. This stem is recognized by RNT1P, an RNase III enzyme, and is essential in the snoRNA maturation pathway. In this thesis, our objective is to characterize this motif and to integrate it in the search for new snoRNAs.

The cleavage motif is already known, from previous studies, to be a stem-loop motif, with a loop of the form DGNN, with the first 3 base pairs being Watson-Crick base pairing. This stem had to be 14-16bp long, because of a cleavage site at one of its end. Based on a set of experimentally validated tetraloop motifs (RNT1P substrates), we used different approaches to get a more specific characterization of the tetraloop : mutual information statistics, pattern matching, secondary structure analysis, classification by a machine learning approach (SVM).

We then present a new method for snoRNA search that integrates this RNT1P cleavage motif. It is based on the search of various conserved primary and secondary structures, on a comparative genomic validation method, and on an efficient clustering algorithm. With this novel approach, we have been able to identify 32 new substrates, 3 of them showing a differential cellular concentration of transcripts, a pattern highly related to snoRNA families. Those are good candidates for experimental validation.

TABLE DES MATIÈRES

RÉSUMÉ . . . . . **iii**

ABSTRACT . . . . . **v**

TABLE DES MATIÈRES . . . . . **vi**

LISTE DES TABLEAUX . . . . . **x**

LISTE DES FIGURES . . . . . **xii**

LISTE DES ANNEXES . . . . . **xvi**

LISTE DES SIGLES . . . . . **xvii**

REMERCIEMENTS . . . . . **xviii**

CHAPITRE 1 : INTRODUCTION **1**

CHAPITRE 2 : CONTEXTE BIOLOGIQUE **5**

2.1 Introduction 5

2.2 Annotation génétique 5

2.3 ADN et ARN 6

2.4 Structure secondaire de l'ARN 7

2.5 SnoRNA 10

2.6 Tetraloop 12

2.7 Concentration cellulaire du transcrit snoRNA 14

2.8 Évolution des espèces du genre *Saccharomyces* 16

2.9 Analyse des résultats biologiques obtenus 16

2.10 Conclusion 18

CHAPITRE 3 : CONTEXTE INFORMATIQUE 19

3.1 Introduction 19

3.2 Recherche de mots dans un texte 19

3.2.1 Recherche exacte d'un mot dans un texte 20

3.2.2 Recherche multiple 21

3.3 Alignement de séquences 22

3.3.1 Distances entre deux séquences biologiques 23

3.3.2 Algorithmes de programmation dynamique pour l'alignement de deux séquences 24

3.3.3 Recherche approchée de mots 26

3.3.4 BLAST 27

3.3.5 Alignement multiple 29

3.4 Conclusion 31

CHAPITRE 4 : MÉTHODES EXISTANTES POUR LA RECHERCHE D'ARN 32

4.1 Introduction 32

4.2 Prédiction de structures secondaires 33

4.2.1 Minimisation de l'énergie libre 33

4.2.2 MFold 34

4.2.3 Vienna RNA Package 34

4.3 Méthodes basées sur l'entraînement 35

4.3.1 Cove 35

4.4 Méthode généraliste 36

4.4.1 RNAMOTIF 37

4.5 Méthodes sur mesure 39

4.5.1 Recherche de snoRNAs de type C/D 40

4.5.2 Recherche de snoRNAs de type H/ACA 41

4.5.3 Recherche de sites de clivage à l'enzyme RNT1P 44

4.6 Conclusion 46

CHAPITRE 5 : NOUVELLE MÉTHODE DE RECHERCHE DE SNORNAS DE TYPE C/D 47

5.1 Introduction 47

5.2 Approche générale 49

5.2.1 Procédure de recherche des séquences génétique de snoRNA . . 50

5.2.2 Procédure de recherche du signal RNT1P 51

5.2.3 Corrélation de distance 55

5.2.4 Procédure de validation par homologie de séquence 56

5.2.5 SVM 57

5.3 Choix des paramètres de la recherche 62

5.3.1 Courbe ROC 62

5.3.2 Choix des paramètres relatifs au motif tetraloop 62

5.3.3 Choix des seuils de l'étape BLAST 65

5.3.4 Validation du modèle SVM 65

5.4 Conclusion 68

CHAPITRE 6 : RÉSULTATS 69

6.1 Introduction 69

6.2 Résultats intermédiaires 69

6.3 Existence de sites de méthylation pour les candidats via snoSCAN . . . 71

6.3.1 Évaluation des candidats en fonction du signal RNT1P 72

6.3.2 Concentration cellulaire des transcrits des candidats finaux . . . 74

6.4 Candidats finaux et annotation des régions sur le génome 76

6.5 Discussion 76

6.5.1 Analyse globale de la méthode présentée 76

6.5.2 Justification de l'approche 78

6.6 Conclusion 80

CHAPITRE 7 : CONCLUSION 81

BIBLIOGRAPHIE **84**

I.1 Vrais snoRNAs xix

I.2 Ensemble d'entraînement pour le SVM xix

I.3 SVM xix

LISTE DES TABLEAUX

3.1 Matrice consensus pour l'alignement présenté à la figure 3.7. 30

5.1 Règles inférées sur le motif RNT1P. Les candidats ne respectant pas celles-ci sont éliminés. W indique une base dans l'hélice, N une base dans la boucle, l'indice se référant à la figure 5.3. 55

6.1 Progression du nombre de FP et TP snoRNAs dans notre méthode. La première colonne indique les principales étapes (voir figure 5.1). Les deux colonnes suivantes sont relatives au nombre de FP et TP snoRNAs respectivement. Vient ensuite la mesure de spécificité puis de sensibilité. 70

6.2 Présentation des 6 candidats (parmi 32) retrouvés par snoSCAN (score supérieur à 12.8). La deuxième colonne donne le score retourné par snoSCAN. La troisième indique le site de méthylation, sous forme simplifiée. Par exemple, le site de méthylation désigné par RDN18-1-Um1289 indique que c'est l'uracile à la position 1289 dans la séquence de la sous-unité ribosomale 18s (petite sous-unité) qui devrait être méthylée. Lorsque le 18 est remplacé par 25, on parle de la grande sous-unité ribosomale. . . . 72

6.3 Ensemble de candidats séparés en fonction du score obtenu par la méthode DRSD (voir section 4.5.3). La section de gauche présente les 16 candidats ayant un score inférieur au seuil, l'autre section présente les 16 dont le score excède le seuil. La première colonne de chaque section indique les noms des candidats, la seconde est relative au programme de recherche de tetraloop développé à Sherbrooke et donne le score de probabilité associé au site trouvé par ma méthode. 73

6.4 Table présentant les 23 vrais snoRNAs dont la tetraloop est clivée expérimentalement (section de gauche) et les 32 candidats finaux (section de droite) en relation avec la concentration cellulaire de transcrit. Les molécules du haut (Snr65 à Snr47 dans la section de gauche, psnr2417 à psnr624 dans l'autre section) montrent une valeur de concentration plus élevée pour le transcrit snoRNA que celui portant le tetraloop, les autres non. La première colonne de chaque section indique le nom du candidat, suivent les deux colonnes relatives au snoRNA, soit la quantité de transcrits et la taille de celui-ci. Les deux dernières colonnes de chaque section concernent le transcrit codant le tetraloop, soit la concentration (conc.) et la taille de celui-ci (transcrit). La résolution pour la taille du transcrit est de 8nt [14]. 75

I.1 Description des vrais snoRNAs utilisés provenant du site web CYGD [19]. La première colonne correspond au nom donné pour la molécule, la seconde au numéro du chromosome, vient ensuite les bornes de la molécule dans le génome, puis la longueur de ces éléments. xx

I.2 Quantités respectives des différentes tiges-boucles utilisées lors de l'entraînement du SVM. xxi

LISTE DES FIGURES

2.1 Exemple d'un ARN de transfert pour la phénylalanine chez la levure (figure 1 de l'article [47]). La séquence est composée de 76nt. Les nucléotides autres que les caractères normaux sont des nucléotides modifiés, par exemple ψ :pseudouridine et m^2G :2'-O-méthylguanosine. 9

2.2 Exemple de différents motifs de structure secondaire (figure du livre [35]). (A) : une tige-boucle, (B) : une boucle interne, (C) : un « bulge » et (D) : une boucle multiple (« multibranch loop »). 9

2.3 Structure consensus du snoRNA de type boîte C/D. 10

2.4 Description d'un motif présentant un « long range interaction » (figure 2 de [18]). Un snoRNA, Snr53, se retrouve à l'intérieur du motif tetraloop. C1, C2 et C3 indiquent les endroits de clivage. 14

2.5 Structure recherchée pour le « tetraloop ». 15

2.6 Modèle de maturation des snoRNAs via la représentation schématique de son transcrit. Le signal de clivage à l'enzyme RNT1P se trouve ici en amont du snoRNA. B et D présentent la concentration de transcrit, mesurée par les micro-puces, en fonction de la position sur la séquence génomique. A) Présentation d'un transcrit typique portant le snoRNA et le motif tetraloop, le site de clivage étant de 14 à 16pb de la boucle. B) Présentation d'un profil pour une cellule Δ RNT, ou lorsque le motif de clivage est absent (la séquence n'est pas clivée). C) Clivage du motif par l'enzyme et dégradation (les portions ombragées) du transcrit. D) Différence de concentration cellulaire entre le snoRNA et son signal tetraloop dû au clivage par RNT1P et de la dégradation subséquente du signal tetraloop. 15

2.7 Arbre phylogénétique des levures *sensu stricto* et *sensu lato*. Le « outgroup » est ici *K.lactis* ou *S.pombe* [13]. 17

3.1 Déroulement de l'algorithme KMP. À chaque position j de T, les caractères de P sont parcourus de gauche à droite tant qu'ils sont identiques aux caractères de T. On s'arrête au premier caractère différent, ici le caractère à la position $j+i$. Le mot est ensuite décalé en fonction de sa périodicité (décalage d calculé au cours de la phase de pré-traitement du mot P), et la recherche reprend à la position $j+d$. 21

3.2 Arbre AC pour l'ensemble $\mathscr{P} = \{abbac, ac, bacd, ababc\}$. La fonction d'échec est représentée par les flèches. 22

3.3 Alignement des deux séquences $\mathscr{S} = TCGC$ et $\mathscr{T} = TACGG$. La distance d'édition entre les deux séquences est 2 (dernière case en bas à droite). Les flèches permettent de retracer un alignement entre les deux séquences. L'alignement correspondant est indiqué à droite de la table de programmation dynamique. 25

3.4 Alignement local de deux séquences pour la distance de similarité suivante : 2 pour une paire identique de nucléotides, et -1 pour une insertion/suppression ou mismatch. Les sous-séquences alignées sont celles entre crochets. La valeur de similarité est de 8. 25

3.5 Fragmentation du mot requête de taille m en ses $m-11$ sous-mots de taille 12. 28

3.6 Exemple d'un HSP et de son élongation. 28

3.7 Exemple d'un alignement multiple de cinq séquences d'ADN et le consensus produit (signature). 30

3.8 Exemple de scores induits par la méthode SP. Le score est exprimé en distance d'édition. 31

4.1 Procédure du programme Cove (figure 3 tirée de [44]). 35

4.2 Structure recherchée, et interprétation de la description par le programme RNAMOTIF. H représente une portion hélicale, avec 5 (respect. 3) représentant le brin 5' (respect. 3'), ss représente une structure simple brin. 37

4.3 Descripteur plus complexe pour RNAMOTIF permettant de retrouver un ARN de transfert (figure tirée de l'article [7]. 39
4.4 Modèle probabiliste développé par Lowe et Eddy (figure 2 de [51]). . . 41
4.5 Description schématique d'un snoRNA de type H/ACA. On remarque les 2 boîtes conservées(H et ACA) ainsi que les séquences guide ψ_3 et ψ_4, qui se retrouvent complémentaires au brin d'ARN ribosomal (figure 1 de [15]). 42
4.6 (A) : consensus recherché pour le site de clivage à l'enzyme RNT1P ; (B) : diagramme de recherche. Figure 1 de [18]. 45

5.1 Procédure globale de recherche élaborée pour trouver de nouveaux snoRNAs. Les nombres sur les flèches indiquent le nombre de candidats snoRNAs restants après l'étape de filtration, ceux écrits à la droite du " :" sont les snoRNAs TP. . 49
5.2 Procédure de recherche des snoRNA de type C/D. 50
5.3 Tige-boucle correspondant au site de reconnaissance de RNT1P (tetraloop). Pour tout indice i, "$W_i - W_i'$" désigne un appariement. 52
5.4 Distribution des scores centrés-réduits pour chaque paire de nucléotides résultant de la statistique d'information mutuelle. Les scores supérieurs à 1,6 sont à la droite de la ligne pointillée. 56
5.5 Méthode de transformation du signal biologique en signal numérique. n= le nombre de bases successives à considérer. Codage des symboles : A=0, C=1, G=2, U=3, (= 1, . = 0 60
5.6 Résultats en spécificité et en sensibilité de la recherche du tetraloop. Chaque point représente une combinaison donnée de tous les paramètres de la recherche. La ligne pleine représente une décision aléatoire, et le rectangle les résultats que l'on juge les meilleurs. Le point sélectionné dans notre recherche est indiqué par la flèche et correspond aux paramètres définis en section 5.2.2. 64

5.7 Courbe ROC permettant de vérifier quel ensemble de paramètres est le plus approprié à notre analyse de candidats en regard de l'analyse SVM. Chaque point correspond aux valeurs de sensibilité et de spécificité associé au vecteur de trois paramètres (n, γ, C). La ligne pleine définie la décision aléatoire, le rectangle est un seuil d'acceptation en terme de sensibilité (78% = $\frac{18}{23}$ TP). Le point choisi est indiqué par la flèche et correspond aux valeurs suivantes : $n = 4, C = 141, \gamma = 2 \times 10^{-4}$. 67

LISTE DES ANNEXES

Annexe I : . . . . . **xix**

LISTE DES SIGLES

ADN	Acide désoxyribonucléique
ARN	Acide ribonucléique
ARNm	ARN messager
ARNt	ARN de transfert
ASE	Antisens element
FN	faux négatif
FP	faux positif
HMM	Hidden Markov Model
INC	Inconnu
indel	insertion / délétion
ORF	Open Reading Frame
NCE	Non Coding Exon
nt	Nucléotide
pb	Paire de Bases
snoRNA	Small Nucleolar RNA
SVM	Support Vector Machine
TN	vrai négatif
TP	vrai positif
WC	Watson–Crick

REMERCIEMENTS

Un gros merci à Nadia pour sa patience et sa rigueur, valeurs qu'elle a essayé de m'inculquer bien malgré moi. Merci encore pour avoir cru en mes capacités, et m'avoir donné la chance d'effectuer des études graduées et de me dépasser dans celles-ci.

Merci à ma famille pour m'avoir encouragé dans les études.

CHAPITRE 1

INTRODUCTION

Le dogme central de la biologie moléculaire a été sérieusement remis en question suite à la découverte de groupes non-codants ayant des fonctions catalytiques (ribozyme [41]). Bien que nous connaissions l'existence d'ARN non-codant depuis la découverte du ribosome et des ARNt, à cette famille fonctionnelle s'est ajoutée plusieurs autres types de molécules (siRNA [3], microRNA [10], snoRNA [1]). Il devient clair que l'ARN n'est pas circonscrit qu'au seul rôle d'intermédiaire entre ADN et protéine, et l'étude des différents types d'ARN et leurs fonctions ajoute une nouvelle dimension à la compréhension de la cellule et du génome. Comme conséquence, la recherche sur l'ARN n'a cessé de s'amplifier ces dernières années.

Plusieurs méthodes bio-informatiques appliquées aux ARN non-codants ont été développées dans cette optique, permettant la caractérisation ou la recherche de familles d'ARN. En particulier, des algorithmes de recherche "sur mesure" ont été développés afin de retrouver automatiquement les structures d'ARN les plus étudiées et les mieux caractérisées. C'est le cas, en particulier, des structures d'ARN de transfert [42, 45], des introns de type I et II [28] et des snoRNAs [15, 51]. Cependant, la majorité des familles d'ARN sont très difficiles à caractériser, et les approches automatisées de recherche permettent, au mieux, d'obtenir des résultats partiels. Dans ce cas, on a plutôt recours à des méthodes plus générales, prenant en entrée une description de la molécule à rechercher, et permettant de retrouver, dans un génome ou une base de données, tous les motifs se conformant à cette description [23]. On peut alors essayer plusieurs descriptions possibles et affiner ainsi la connaissance que l'on a de la famille d'ARN. Les méthodes d'apprentissage sont également de plus en plus utilisées afin de générer de l'information sur des classes de molécules [44]. Ces méthodes, à la croisée des chemins entre les mathématiques, les statistiques et l'informatique, permettent de cibler l'information pertinente à la classification d'un ensemble de données.

Parmi les ARN non-codants, les snoRNAs sont retrouvés chez plusieurs espèces

d'eucaryotes. Il en existe plusieurs familles ayant des fonctions différentes. Les caractéristiques principales de cet ensemble de molécules sont leur localisation dans la cellule (au niveau du nucléole), leur interaction avec des protéines spécifiques ayant pour fonction la modification de l'ARN et leur fonction de guide pour ces mêmes protéines.

Dans ce mémoire, nous nous intéressons à la famille des snoRNAs de type C/D. Ils sont impliqués principalement dans la méthylation de certains nucléotides sur les ARN ribosomaux. Les snoRNAs de type C/D font partie des "nouveaux" ARN fonctionnels les plus étudiés. La famille a été caractérisée par des conservations de séquences et de structures [1, 12, 18, 25, 30, 34, 51], et des méthodes de recherche "sur mesure" ont été développées. En particulier, l'approche de T.Lowe *et al.* [51] avait permis de trouver 22 nouveaux snoRNAs de ce type chez *Saccharomyces carlsbergensis*, une levure proche parente de *Saccharomyces cerevisiae*, laissant au plus 4 snoRNAs non découverts méthylant l'ARNr [51]. De plus, il pourrait exister des snoRNAs méthylant d'autres types d'ARN, ou des snoRNAs à l'état de pseudogènes ne pouvant pas être détectés par les méthodes existantes.

Dans ce mémoire, notre objectif est de développer une nouvelle approche bio-informatique permettant de trouver des snoRNAs de type C/D non encore découverts par les méthodes précédentes. Pour ce faire, nous considérons une contrainte biologique qui n'a pas été prise en compte par les approches antérieures : la présence, en amont ou en aval de la séquence génomique du snoRNA, d'un motif structural représentant le signal de reconnaissance de l'enzyme de clivage RNT1P. Le logiciel de référence pour la recherche de snoRNA de type boîte C/D, snoSCAN [51], bien qu'ayant permis de découvrir une bonne partie de ces snoRNAs ayant un site de méthylation sur l'ARN ribosomal, est limité à cette seule sous-famille de snoRNA de type C/D. Cette approche est restrictive puisque l'on sait que d'autres ARN peuvent aussi posséder des nucléotides méthylés par des snoRNAs de types C/D [24]. La démarche présentée en [51] exclut aussi que de nouvelles fonctions des snoRNAs puissent exister. Il est alors essentiel de disposer d'une contrainte alternative permettant d'obtenir des candidats prometteurs, qui pourront finalement être testés en laboratoire.

Le site RNT1P de reconnaissance à l'enzyme RNT1P a été étudié par l'équipe de S.

Abou Elela [8, 21, 22] et celle de G.Chanfreau [26, 29, 30]. Une étude de cette littérature, de même que plusieurs entretiens avec S. Abou Elela nous a mis sur la piste d'un motif tige-boucle, la boucle étant de type DGNN, avec pour seules contraintes sur l'hélice les 3 premiers appariements proximaux à la boucle, qui devaient être de type WC. Une grande partie de ce projet a consisté à trouver des façons de mieux caractériser ce motif, toujours dans le but de trouver des snoRNAs de type C/D non répertoriés.

Le travail a porté sur le génome de *Saccharomyces cerevisiae*. La méthode développée retrouve 32 nouveaux candidats snoRNAs étant sous le contrôle de maturation de l'enzyme RNT1P dont 3 semblent très prometteurs ; certains de ceux-ci pourraient se révéler être de nouveaux snoRNAs lorsque validés en laboratoire.

Dans le chapitre 2, je présenterai les concepts biologiques qui seront l'objet de mon mémoire. Je parlerai entre autres des snoRNAs de type C/D, du motif tetraloop, d'annotation de génome, et du groupe *Saccharomyces*.

Le chapitre 3 servira à décrire les principaux algorithmes utiles à la compréhension de mon sujet de recherche. Les concepts de « pattern matching » et d'alignement de séquence, ainsi que l'idée générale de Blast, une heuristique pour la recherche approchée de séquence dans des bases de données, seront présentés.

Le chapitre 4 présente l'état actuel de la recherche en bio-informatique, et plus spécifiquement au champ relié à l'étude des ARN non-codants. Je présenterai 3 méthodes appliquées à la recherche de snoRNAs et une permettant de retrouver les tetraloops associés aux snoRNAs. Je discuterai de repliement de structure, en introduisant deux des logiciels les plus utilisés (mFold et Vienna RNA Package). Cove sera présenté, permettant d'en apprendre davantage sur les méthodes d'apprentissage. Enfin, on verra comment fonctionne RNAMOTIF, une méthode générale utilisée par les biologistes.

La méthode développée sera ensuite élaborée au chapitre suivant. L'approche étant principalement un agencement de filtres sur les snoRNAs de type C/D, chacun de ceux-ci sera vu en détail. On verra aussi comment paramétrer les différents filtres afin d'obtenir les résultats les plus "appropriés".

Au chapitre 6, nous présenterons nos résultats via la méthode de T.Lowe [51], la méthode développée par l'équipe de S. Abou Elela [18], et en analysant la concentration

de transcrit snoRNAs. Nous discuterons aussi des défauts de notre approche.

Je conclurai au chapitre 7 en soulignant les réalisations de mon approche.

CHAPITRE 2

CONTEXTE BIOLOGIQUE

2.1 Introduction

J'introduis dans ce chapitre plusieurs notions biologiques utilisées dans le cadre de ma recherche. Je parlerai tout d'abord de l'annotation génétique, je discuterai ensuite de l'ADN et l'ARN au sens large, puis du repliement de l'ARN. J'introduirai ensuite deux familles de molécules d'ARN qui sont de première importance dans ma recherche. Je discuterai ensuite d'une étude ayant été faite sur le génome de *Saccharomyces cerevisiae*, puis du groupe *Saccharomyces*. Finalement, j'introduirai certaines définitions utiles lors de l'analyse de mes prédictions.

2.2 Annotation génétique

Un des champs d'étude de la bio-informatique est l'annotation de génome. Cette procédure consiste à regrouper des portions de génome ayant des fonctions similaires, donc faire une classification. Cette méthode permet d'ajouter de l'information, utilisée, par exemple, dans une analyse de candidats provenant d'une méthode de prédiction de promoteurs de gènes. Dans cet exemple, on peut se limiter aux régions en amont ou en aval de gènes exprimés, donc aux UTR (UnTranslated Region). Plusieurs régions génétiques existent, je définis subséquemment les régions utilisées dans le cadre de ma recherche :

- *ORF* : Open Reading Frame. Toute séquence d'ADN ou d'ARN qui peut être traduite en une protéine. Définie généralement comme la portion de la séquence partant du codon d'initiation, et se terminant au codon stop.
- *Exon* : Portion du transcrit d'ARN qui, suite à l'épissage, reste dans le transcrit. Ces sous-sections de la séquence sont donc responsables du codage de la protéine.
- *Gène non-codant* : Gène sous le contrôle d'un promotteur permettant son expression, mais qui ne possède pas de ORF (Open Reading Frame). Ce gène ne code

donc pas pour une protéine.

- *Polycistron* : ARNm contenant l'information génétique pour plusieurs molécules (protéine, ARN fonctionnels). Retrouvé principalement chez les procaryotes, mais les snoRNAs ont cette organisation chez la levure.
- *Intron* : Segments de l'ARN qui sont soustraits de la séquence lors de l'épissage. Ils ne participent pas à la traduction en protéines.
- *UTR* : UnTranslated Region. Région non-traduite. Région de l'ARNm en amont ou en aval de la séquence codante, qui va généralement contenir des promoteurs ou des éléments modifiant le degré de traduction en protéines de la séquence.

2.3 ADN et ARN

Dans une cellule, qu'elle soit eucaryote ou procaryote, il existe plusieurs types de biopolymères : ADN, ARN et protéines pour ne nommer que ceux-ci. Une séquence d'ADN est une suite de quatre nucléotides : adénine, cytosine, guanine et thymine, représentés schématiquement par leurs bases azotées : A, C, G, T. Les composantes principales d'un nucléotide sont : un sucre (le désoxyribose dans le cas de l'ADN et le ribose dans l'autre), et une base azotée. La fonction de l'ADN est de stocker l'information génétique de la cellule. Toutes les fonctions d'une cellule, de même que certains aspects physiologiques, sont dérivées de cette information génétique. Chez les eucaryotes, l'ADN est circonscrit au noyau, dans les mitochondries et les chloroplastes. Afin que les fonctions cellulaires soient préservées, l'ADN doit être transcrit en une autre macro-molécule appelée ARN. L'ARN se différencie de l'ADN par le fait qu'elle possède un ribose au lieu d'un désoxyribose, et que la thymine (T) est remplacée par l'uracile (U). On parle de biopolymère d'ADN ou d'ARN car une chaîne moléculaire peut être construite lorsque l'on raccorde les nucléotides entre eux par des liens 3' phosphodiester, via une enzyme appelée ARN ou ADN polymérase.

L'ARN joue un rôle capital dans la cellule : il sert d'intermédiaire entre l'ADN et les molécules effectives que sont les protéines. Il agit donc comme le véhicule de l'information génétique. L'information contenue dans sa chaîne permet un arrangement très

précis des acides aminés dans la protéine. C'est pourquoi nous appelons ce type de molécule ARN messager (ARNm). Il existe plusieurs raisons à l'utilisation d'une molécule intermédiaire entre notre génome, encodé dans l'ADN, et les protéines : la régulation de la quantité de protéines en est une, la protection de l'ADN contre les mutations et délétions en est une autre. Une troisième raison primordiale est que l'ARN n'est pas circonscrit qu'au seul rôle de messager. En effet, il existe d'autres familles d'ARN dits "non-codants". En particulier, le ribosome, cette machine biologique impliquée dans la fabrication des protéines, est composé à 66% d'ARN. Plus récemment, on a aussi retrouvé de l'ARN qui pouvait agir comme une enzyme en ayant un pouvoir catalytique [41]. L'ARN peut même adopter des structures conformationnelles les rendant aptes à être reconnues par des enzymes de clivage, permettant ainsi de l'épissage alternatif, et un meilleur contrôle de la transcription. Certains auteurs ont repris le concept de "monde d'ARN" de Gilbert [53], non pas pour expliquer expliquer les origines de la vie via ces nouveaux rôles de l'ARN, mais bien pour caractériser ces fonctions complexes, et montrer l'importance de l'ARN dans la cellule.

L'ARN messager code pour les protéines. Chaque triplet d'une séquence codante est appelée "codon" et code pour un acide aminé particulier. On dit qu'il y a dégénérescence du code génétique car on a 4^3=64 codons différents mais seulement 20 acides aminés ; un même acide aminé est donc codé par plusieurs codons différents. Généralement, une traduction (transposition codon vers biopolymère d'acides aminés rendu possible par le ribosome) s'effectue lorsque le codon méthionine (AUG) est rencontré. Le ribosome va ensuite traduire la suite de codons jusqu'au codon stop (UGA, UAG, UAA).

2.4 Structure secondaire de l'ARN

L'ARN se replie dans l'espace grâce à des appariements (ou paires de bases, notées *pb* dans le texte) entre les nucléotides, les plus stables étant les appariements Watson-Crick (WC) : (A,U) et (C,G). D'autres appariements, dits non-canoniques, peuvent également se former, le plus fréquent étant l'appariement (G,U) appelé wobble. Ces appariements sont rendus possibles grâce à des interactions appelées ponts hydrogènes.

Elles impliquent des donneurs d'hydrogènes, comme les groupements NH ou NH_2 et des groupements accepteurs comme N ou O. Ces interactions transforment la structure linéaire du biopolymère en une structure en trois dimensions. La structure secondaire est une représentation schématique du repliement 3D de la molécule, intermédiaire entre la structure linéaire et tertiaire. Elle tient compte de la nature des nucléotides impliqués, mais ne permet pas de savoir quelles faces des nucléotides se retrouvent exposés au solvant. Je donne ci-dessous une définition rigoureuse généralement utilisée de la structure secondaire d'ARN.

Soit $S = s_1 \cdots s_n$ une séquence d'ARN (suite de A, C, G, U). On note (s_i, s_j) un appariement entre deux bases s_i et s_j. Une structure secondaire $\mathscr{S}$ de S est une suite d'appariements et de bases non appariées, vérifiant les conditions suivantes :

1. Pas de chevauchement : si s_i est appariée avec s_j, alors s_i n'est appariée avec aucune autre base. Autrement dit $\mathscr{S}$ ne peut pas contenir deux appariements différents (s_i, s_j) et (s_i, s_k).
2. Pas de pseudo-noeuds : pour tout indice $h < i < j < k$, $\mathscr{S}$ ne peut pas contenir à la fois (a_h, a_j) et (a_i, a_k).
3. Pas de tournants brusques : si (a_i, a_j) est dans $\mathscr{S}$, alors $|j - i| \geq 3$. Autrement dit, une boucle finale contient au moins 3 nucléotides.

En réalité des structures d'ARN peuvent contenir des chevauchements ou des pseudo-noeuds. Cependant de telles interactions sont généralement considérées comme des interactions tertiaires. Un exemple de structure secondaire est représenté à la figure 2.1.

Les différentes méthodes de caractérisation, de repliement et de recherche de structures secondaires nécessitent de subdiviser la structure en un ensemble de sous-structures. La structure élémentaire la plus connue est la tige-boucle, formée d'une hélice et d'une boucle (qui doit être d'au moins 3 nucléotides conformément à la contrainte 3 ci-dessus). Les autres structures élémentaires les plus utilisées sont les « bulges », les boucles internes et les boucles multiples. Ces différentes sous-structures sont illustrées à la Figure 2.2.

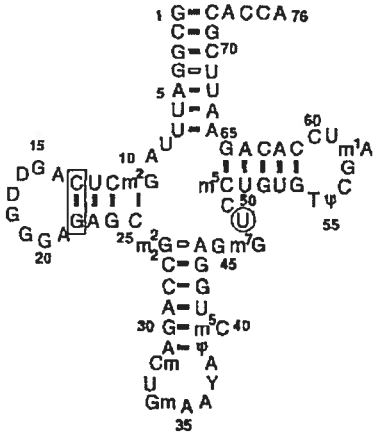


Figure 2.1 – Exemple d'un ARN de transfert pour la phénylalanine chez la levure (figure 1 de l'article [47]). La séquence est composée de 76nt. Les nucléotides autres que les caractères normaux sont des nucléotides modifiés, par exemple ψ :pseudouridine et m^2G :2'-O-méthylguanosine.

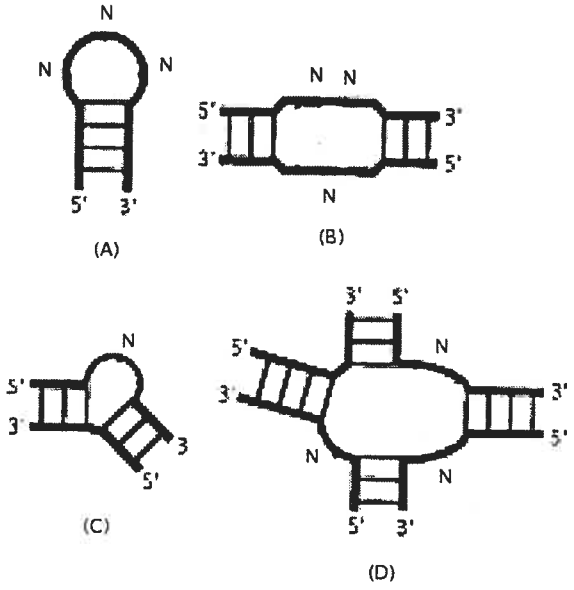


Figure 2.2 – Exemple de différents motifs de structure secondaire (figure du livre [35]). (A) : une tige-boucle, (B) : une boucle interne, (C) : un « bulge » et (D) : une boucle multiple (« multibranch loop »).

2.5 SnoRNA

Chez *Saccharomyces cerevisiae*, la plupart des snoRNAs sont transcrits comme des unités indépendantes (gène non-codant) ou en transcrits polycistroniques. Seulement 7 des 66 snoRNAs connus sont dans des introns de gènes codant pour les protéines ribosomales, confirmant indirectement dans ces cas-là leur association avec le ribosome [18]. Un snoRNA est un ARN qui se retrouve dans le nucléole, complexé avec des protéines pour former le snRNP (« Small Nucleolar RiboNucleoParticle) ». Le nucléole est un organelle multifonctionnel du noyau intervenant dans la maturation de différentes classes d'ARN cellulaires [34]. À ce jour, on peut distinguer trois classes de snoRNA : la famille MRP/7-2 avec, pour seule molécule, MRP/7-2, la famille à boîtes C/D, et la famille à boîtes H/ACA.

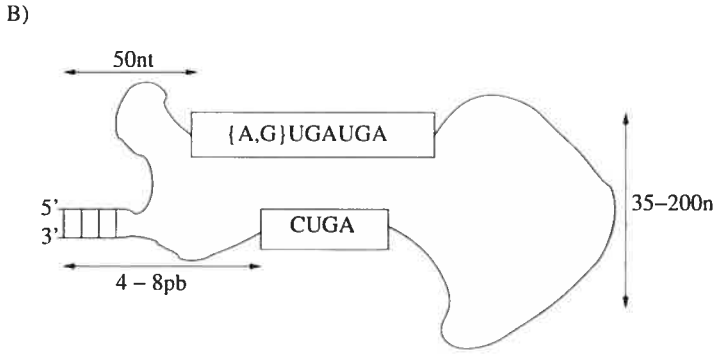


Figure 2.3 – Structure consensus du snoRNA de type boîte C/D.

Dans le cadre de ce mémoire, les snoRNAs qui nous intéressent sont les snoRNAs de type C/D. Ils sont impliqués principalement dans la méthylation de certains nucléotides sur les ARNs ribosomaux et les snRNAs. Leur séquence a une taille variant de 70 à 250 nucléotides, et est caractérisée par deux motifs conservés, soit la boîte C (consensus $\{A,U\}UGAUGA$) et la boîte D (consensus $CUGA$), ainsi que par une hélice terminale qui ferme la molécule (voir la Figure 2.3). Selon les données d'alignement obtenues du site web de T.Lowe [4], 122 snoRNAs sur 236 possèdent exactement le consensus de la boîte C (181 sur 236 pour la boîte D). Une autre caractéristique commune est la distance inter-boîtes ; elle varie de 35 à 200 nucléotides, et peut contenir des copies imparfaites

(avec erreurs) des boîtes C et D, appelées C' et D' [51]. Le dernier élément d'intérêt est le motif antisens (ASE), qui est complémentaire à la zone de méthylation sur l'ARN cible. Le motif antisens sert à positionner la boîte D du snoRNA à 5 nucléotides de la base à méthyler sur l'ARN cible [12], permettant aux protéines accessoires de méthyler celui-ci. Le motif ASE se retrouvera donc toujours directement en 5' de la boîte D ou D'.

Les snoRNAs ne semblent pas adopter de structure secondaire précise. Les motifs conservés de cette famille sont requis pour la localisation de la molécule dans le noyau, son accumulation et son association avec le complexe de particules formant le snRNP [51]. Ce sont ces caractéristiques des snoRNAs qui sont à la base des méthodes existantes de recherche automatique.

La méthylation est un traitement post-transcriptionnel fréquemment observé sur certaines séquences de nucléotides. Les snoRNAs de type C/D sont impliqués dans la 2'−O−méthylation, caractérisée par l'ajout d'un groupement méthyl à l'oxygène annoté 2'.

Dans le génome de la levure *Saccharomyces cerevisiae*, le nombre de sites de méthylation sur l'ARN ribosomal est de 55, 51 de ceux-ci étant associés à 41 snoRNAs [51]. Toutefois, à l'heure actuelle, on dénombre 46 snoRNAs de type C/D chez *Saccharomyces cerevisiae* [19]. Ces snoRNAs sont donc responsables de la méthylation d'autres type d'ARNs.

Chez l'homme, le nombre de nucléotides 2'−O−méthylés est de 100. Présentement, 50 snoRNAs de type C/D [32] ont été identifiés. Une étude récente suggère que 202 rétrogènes de snoRNAs de type H/ACA seraient présents chez l'homme [54]. Les rétrogènes sont des fragments d'ADN répétitifs qui sont insérés dans les chromosomes après une rétrotranscription de n'importe quelle molécule d'ARN [36]. Cette étude souligne le fait que le nombre de rétrogènes snoRNA chez l'homme peut être beaucoup plus grand qu'on ne le soupçonne et pourrait nous forcer à revoir notre façon de créer des modèles de recherche. Aucune étude concernant les rétrogènes de snoRNAs de type C/D chez l'homme n'a été publiée à ce jour.

2.6 Tetraloop

Une autre particularité commune à pratiquement tous les snoRNAs, que ce soit de type C/D ou H/ACA, est la présence, en amont ou en aval de la séquence transcrite, à une distance variant de 100 à 200nt [30]*etal*, d'une séquence qui se replie pour former un motif en forme de tige-boucle (figure 2.5), que l'on appellera « tetraloop ». C'est un élément régulateur qui intervient dans le processus de maturation des snoRNAs. Il est reconnu par un enzyme de clivage particulier. Dans le génome de *Saccharomyces cerevisiae* cet enzyme est appelé RNT1P. On sait, d'après Chanfreau *etal* [30], que la voie de transformation associée à RNT1P doit exister chez les autres levures , mais on ne sait pas si cette voie est conservée dans d'autres règnes du vivant. Par contre, on sait qu'il existe des snoRNAs et un homologue de l'enzyme RNT1P chez l'humain.

Voici quelques définitions utiles à la compréhension de la suite.

- *RNase* : Type d'enzyme qui catalyse le clivage d'ARN. Il en existe trois types (I, II et III). La RNase III est une famille regroupant des endonucléases qui clivent l'ARNr (petite et grande sous-unités) de transcrits eucaryotiques.
- *Endonucléase* : Enzyme qui clive à l'intérieur d'une séquence d'ADN ou d'ARN, parfois grâce à la reconnaissance d'un motif particulier.
- *Exonucléase* : Enzyme qui clive 1nt à la fois à une extrémité libre d'ADN ou d'ARN.

Le RNT1P est le seul représentant de la famille des RNases III chez *Saccharomyces cerevisiae* [26]. Bien que la plupart des Rnases III bactériennes et eucaryotiques clivent l'ARN de façon non-spécifique, la présence de la tetraloop se terminant par une boucle AGNN est un déterminant fort pour l'appariement et le clivage par RNT1P [26]. Les enzymes qui se rapprochent le plus chez l'humain de RNT1P sont DICER et DROSHA. La première est localisée dans le cytosol alors que la seconde est dans le noyau. DROSHA est impliquée dans la maturation de microRNA de la même façon que l'implication de RNT1P dans la maturation des snoRNAs chez la levure [5, 38, 43].

Un site de reconnaissance à l'enzyme RNT1P est une tetraloop avec le motif (AGNN) suivie d'une hélice de taille variable, mais dont les 3 premiers appariements sont de type

Watson-Crick. Ce motif est reconnu par l'enzyme RNT1P, qui vient s'y fixer, via son dsRBD (« Double Stranded RNA Binding Domain », ou domaine de liaison à l'ARN double brin) pour ensuite cliver le transcrit d'ARN. Cette molécule sera ensuite modifiée à ses extrémités 3' et 5' via des exonucléases [30](modification post-transcriptionnelle) pour devenir un snoRNA mature.

Toutes les enzymes de type RnaseIII requièrent 12-16 pb (environ un à un tour et demi d'une hélice d'ARN de type A) pour se lier à leur substrat. De plus, l'étude menée par Lamontagne *et al.* [21] démontre que la distance minimale de clivage serait de 14 pb à partir de la tetraloop, c'est-à-dire à 35.42 Ångström (2,53 Ångström d'élévation par pb). Le modèle de recherche doit donc pouvoir chercher des motifs qui pourraient avoir des bulges dans la boucle, permettant ainsi d'avoir la même distance entre le site de clivage et la boucle, tout en ayant des nucléotides libres. De plus, des interactions longue portée (« long range interaction ») ont été observées dans certains motifs [18]. De telles interactions se produisent lorsqu'une portion de la séquence sort de la conformation hélice, et forme une boucle multiple (diagramme (D) de la figure 2.2). On obtient donc une hélice perpendiculaire à l'hélice principale. Cette portion de la structure secondaire peut être longue de plusieurs dizaines de bases, comme le montre la figure 2.4.

Les récentes études sur le site de reconnaissance semblent indiquer que ce soit une conservation de la structure de l'hélice et non une conservation dans la séquence qui soit à l'origine de la reconnaissance par l'enzyme. Même le G central dans la tetraloop, qui semble universel, peut être remplacée par une adénine et le motif peut quand même être reconnu par l'enzyme [18]. La gamme des substrats de cette enzyme est donc large, et il est difficile de cerner les composantes communes.

Dans le cadre de ce mémoire, nous nous concentrerons sur les sites RNT1P présentant une boucle DGNN, D étant le symbole pour tous les nucléotides sauf la cytosine, et nous étudierons la façon de les caractériser afin de mieux cerner les occurrences en relation avec les snoRNAs. Le choix particulier de la boucle est dû au fait qu'aucun substrat connu n'était de type CGNN. On met donc de côté les concepts de NRNN tetraloop, R étant le symbole pour les purines (adénine et guanine), ainsi que de « long range interaction », car ceux-ci introduiraient trop de bruit dans nos résultats. De plus, comme il sera

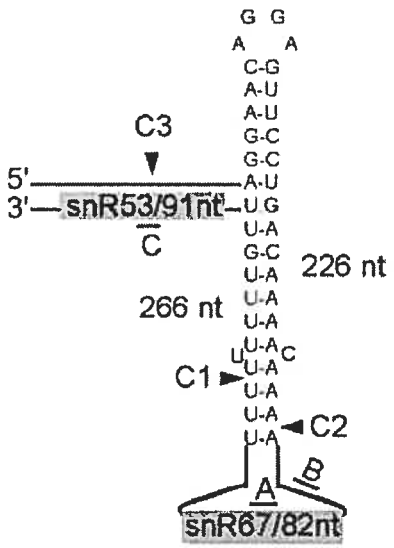


Figure 2.4 – Description d'un motif présentant un « long range interaction » (figure 2 de [18]). Un snoRNA, Snr53, se retrouve à l'intérieur du motif tetraloop. C1, C2 et C3 indiquent les endroits de clivage.

présenté en section 5.3, rechercher une hélice de la même taille que le motif dans la littérature (16pb, de la boucle au site de clivage), introduit trop de bruit dans notre approche. Une taille inférieure (8-9pb) nous permet d'avoir un meilleur contrôle sur les candidats finaux. Nous souhaitons pouvoir trouver de nouveaux snoRNAs, quitte à perdre un peu en sensibilité.

2.7 Concentration cellulaire du transcrit snoRNA

Selon l'équipe de S. Abou Elela, 44 des 66 snoRNAs connus sont traités durant leur phase de maturation par l'enzyme RNT1P [18]. Le modèle général à ce traitement implique le motif de clivage à l'enzyme, et le transcrit snoRNA produit dans une cellule Δ RNT (l'enzyme RNT1P est non-fonctionnelle) présente une taille différente d'une cellule sauvage. La figure 2.6 présente ce profil. La façon de visualiser cette information consiste à utiliser des données de « tiling array ».

Un « tiling array » est une micro-puce permettant de mesurer la concentration de transcrits de gènes à un temps donné dans une cellule [14]. Cette micro-puce contient

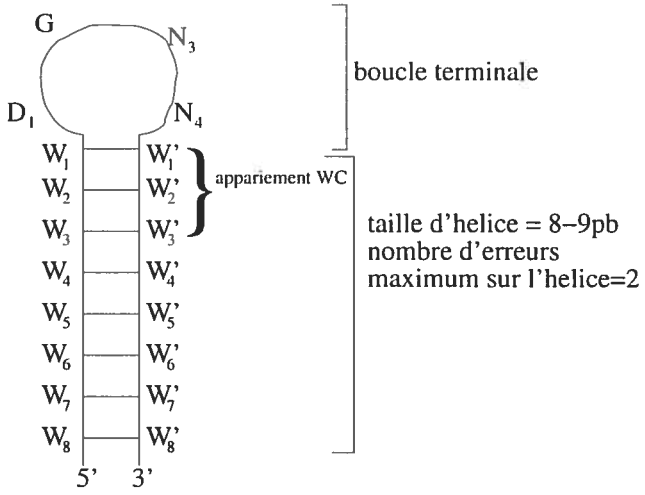


Figure 2.5 – Structure recherchée pour le « tetraloop ».

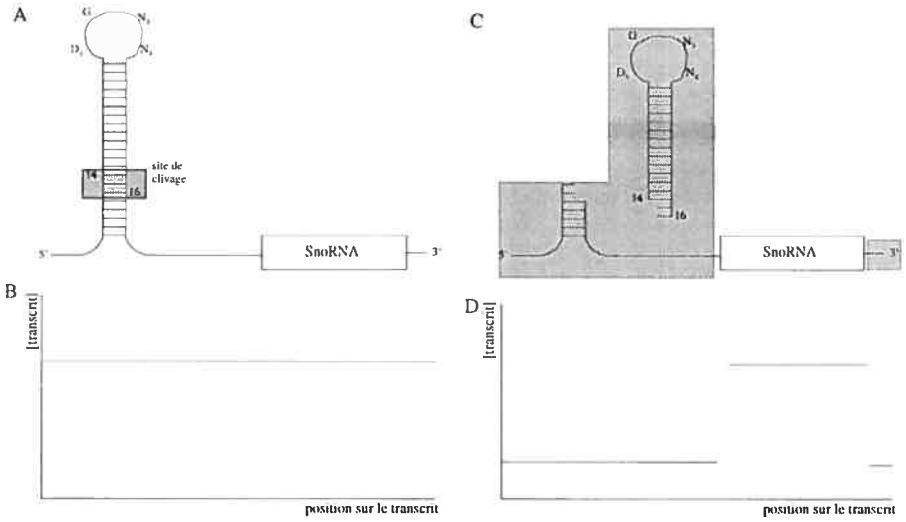


Figure 2.6 – Modèle de maturation des snoRNAs via la représentation schématique de son transcrit. Le signal de clivage à l'enzyme RNT1P se trouve ici en amont du snoRNA. B et D présentent la concentration de transcrit, mesurée par les micro-puces, en fonction de la position sur la séquence génomique. A) Présentation d'un transcrit typique portant le snoRNA et le motif tetraloop, le site de clivage étant de 14 à 16pb de la boucle. B) Présentation d'un profil pour une cellule Δ RNT, ou lorsque le motif de clivage est absent (la séquence n'est pas clivée). C) Clivage du motif par l'enzyme et dégradation (les portions ombragées) du transcrit. D) Différence de concentration cellulaire entre le snoRNA et son signal tetraloop dû au clivage par RNT1P et de la dégradation subséquente du signal tetraloop.

6,25 millions de sondes, la taille de chacune d'entre-elles étant de 25 nucléotides. La configuration de séquence des sondes leurs permet de s'accrocher à la cible à une distance moyenne inter-sonde de 8nt, donnant ainsi une résolution de 8nt. Il s'agit d'une nouvelle approche employée en biologie afin de mieux caractériser le génome d'un organisme, et permet de détecter les régions codantes, les isoformes (produits par l'épissage alternatif). La différence en concentration de transcrits observée en portion D de la figure 2.6, est basée sur les données d'expériences de ces micro-puces. Ce concept sera abordé à la section 6.3.2.

2.8 Évolution des espèces du genre *Saccharomyces*

Comme il a été expliqué en section 2.2, il existe plusieurs types de régions sur le génome. Certaines régions, qu'on peut qualifier de fonctionnelles puisqu'elles contiennent de l'information vitale à la cellule (exon, unité polycistroniques, gène non-codant), subissent une pression sélective purificatrice, c'est-à-dire que des mutations survenant dans ces régions peuvent entraîner la mort de l'organisme.

Quatre levures ont été sélectionnées pour pouvoir faire de la génomique comparative sur mes candidats snoRNAs. Trois d'entre-elles font parti du groupe *Saccharomyces sensu stricto* et sont très proches évolutivement à *Saccharomyces cerevisiae*. On observe l'ordre suivant d'ancêtre commun le plus récent avec *Saccharomyces cerevisiae* : *Saccharomyces mikatae*, *Saccharomyces kudriavzevii*, *Saccharomyces bayanus* (voir figure 2.7). La dernière levure utilisée, *Naunovia castellii*, est considérée comme la levure « outgroup » puisqu'elle fait partie du groupe *Saccharomyces sensu lato*.

2.9 Analyse des résultats biologiques obtenus

Nous concluons ce chapitre en introduisant des concepts classiques généralement utilisés en bio-informatique afin d'évaluer la pertinence de résultats biologiques obtenus par une méthode de recherche. Étant donné que ces concepts seront employés pour évaluer tantôt les candidats snoRNAs obtenus, tantôt les sites de reconnaissance à l'enzyme RNT1P, la définition est faite dans l'absolu, et leur utilisation est comprise dans le

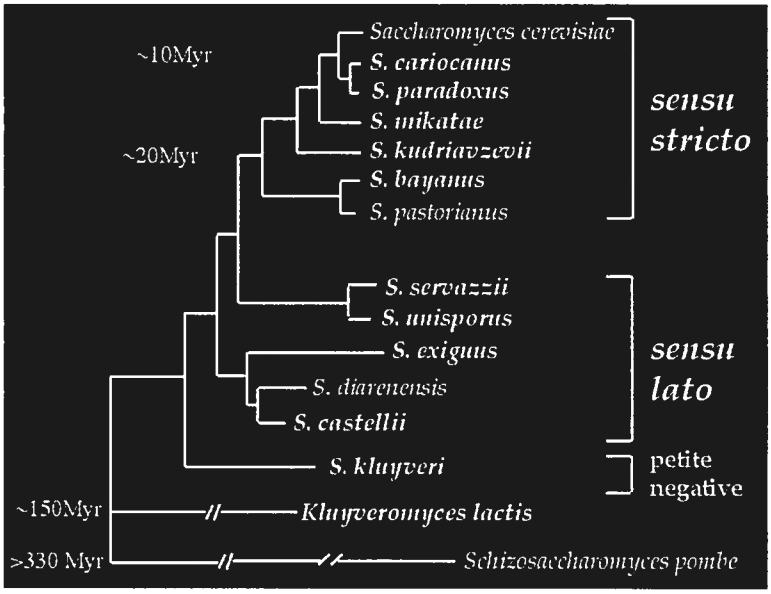


Figure 2.7 – Arbre phylogénétique des levures *sensu stricto* et *sensu lato*. Le « outgroup » est ici *K.lactis* ou *S.pombe* [13].

contexte employé.

Supposons que l'on veuille évaluer une méthode $\mathscr{M}$ de recherche de tous les représentants d'une famille $\mathscr{F}$ de motifs biologiques (famille de gènes, de protéines ou d'ARN particuliers) dans une banque de données biologiques $\mathscr{B}$. L'évaluation de la méthode s'effectue en considérant les notions suivantes :

- *Vrais positifs* (TP) : Séquences reconnues comme appartenant à la famille donnée par des études antérieures et/ou grâce à une validation expérimentale (par exemple, site RNT1P clivé par l'enzyme). Séquences reconnues par la méthode $\mathscr{M}$.
- *Vrais négatifs* (TN) : Séquences non répertoriées dans les bases de données (pas reconnues comme appartenant à la famille $\mathscr{F}$ par des méthodes antérieures), et qui ne sont pas non plus retrouvées par la méthode de recherche en cours d'évaluation.
- *Faux positifs* (FP) : Séquences retrouvées par la méthode $\mathscr{M}$, mais qui ne sont pas reconnues par les méthodes antérieures (non répertoriées).
- *Faux négatifs* (FN) : Séquences répertoriées dans les bases de données mais qui ne sont pas retrouvées par la méthode $\mathscr{M}$.

- *Sensibilité* : Se définit comme étant la fraction du nombre de vrais positifs sur le total des positifs d'une expérience, qu'ils soient retrouvés ou non (FN) : $\frac{TP}{TP+FN}$. Permet de mesurer, lorsqu'utilisée de concert avec la spécificité, si la méthode est fiable, c'est-à-dire qu'elle retrouve bien le type de molécule recherché.
- *Spécificité* : Se définit comme étant la fraction du nombre de vrais négatifs d'une méthode sur la somme des candidats possibles : $\frac{TN}{TN+FP}$. Cette mesure reflète la qualité d'un modèle à ne retourner que ce qui fait vraiment partie de la classe de molécule étudiée. Ce résultat varie entre 0 et 1, plus la valeur tend vers 1, plus le modèle est restrictif à la classe de molécule.

Lorsque l'on développe une méthode bio-informatique ayant pour objectif la caractérisation d'une molécule et sa recherche sur un génome, il est important d'avoir des outils permettant de mesurer la justesse de ses prédictions. Un rapport existe entre la spécificité et la sensibilité et, lorsque les bonnes conditions sont atteintes, on peut avoir confiance en nos résultats. Dans la suite de ce mémoire, notre objectif est d'être le plus spécifique possible, tout en gardant une sensibilité supérieure à 0,7.

2.10 Conclusion

J'ai introduit dans ce chapitre plusieurs notions essentielles à la compréhension de mon domaine de recherche. Je présente au chapitre suivant les composantes informatiques qui ont été utiles dans ce cadre de recherche.

CHAPITRE 3

CONTEXTE INFORMATIQUE

3.1 Introduction

La méthode que nous avons développée pour la recherche de snoRNA nécessite la mise en oeuvre de plusieurs algorithmes différents permettant d'effectuer plusieurs sous-tâches fondamentales, dont l'alignement de séquences, la recherche simple et multiple de motifs, le repliement de structures secondaires et l'analyse phylogénétique. Pour une tâche donnée, la motivation derrière le choix d'un algorithme reste principalement sa complexité en temps, c'est-à-dire le nombre d'opérations qu'une procédure devra effectuer en fonction de l'entrée.

Ce chapitre introduit les différents algorithmes que nous utiliserons dans l'élaboration de notre méthode (voir chapitre 5). Je discuterai tout d'abord d'algorithmes pour la recherche exacte de mots (suite de lettres) dans une base de données. J'introduirai ensuite le problème de l'alignement de séquences génomiques incluant l'alignement global et local de deux séquences ainsi que l'alignement multiple. Dans le cas de l'alignement de deux séquences, je présenterai les algorithmes classiques de programmation dynamique utilisés.

3.2 Recherche de mots dans un texte

La méthode élaborée pour la recherche de snoRNA nécessite l'utilisation d'algorithmes pour la recherche de motifs à différentes étapes, en particulier pour la recherche des boîtes *C* et *D* des snoRNAs, la recherche de séquences complémentaires pouvant former l'hélice terminale du snoRNA ou de l'hélice du site de reconnaissance de l'enzyme RNT1P, ainsi que pour la recherche de séquences homologues à un candidat snoRNA dans différents génomes.

La recherche de motifs est un champ de recherche très vaste, connu également sous le nom de «Pattern Matching». Sans chercher à être exhaustif, j'introduirai brièvement

les différents problèmes ainsi que les algorithmes que nous avons utilisés. Bien que les seules séquences qui nous intéressent dans ce mémoire soient les séquences génomiques définies sur l'alphabet $\{A,C,G,T\}$, nous considérons dans cette section un alphabet Σ quelconque.

Nous avons besoin de quelques définitions préalables. Étant donné un mot $P = P_1 \cdots P_m$ de taille m, un « préfixe » de P est un sous-mot de P de la forme $p_1 \cdots p_i$ avec $2 \leq i \leq m$. Un « suffixe » de P est un sous-mot de P de la forme $p_i \cdots p_m$ avec $1 \leq i \leq m-1$.

3.2.1 Recherche exacte d'un mot dans un texte

Étant donné un texte T de taille n et un mot P de taille m qui sont deux suites de lettres sur un même alphabet Σ, le problème est de trouver toutes les occurrences de P dans T.

L'algorithme naïf consiste à parcourir le texte position par position, et à chaque position comparer chaque caractère du texte avec chaque caractère du mot. La complexité en temps dans le pire des cas de l'algorithme naïf est en $O(mn)$. Bien que cet algorithme soit, en pratique, très rapide pour des textes de taille raisonnable, dans le cas de la recherche dans les banques de données biologiques, il est primordial d'avoir des algorithmes encore plus performants.

Différentes méthodes d'optimisation existent pour la recherche exacte de mots. Elles consistent à effectuer un pré-traitement préalable du mot à rechercher. L'objectif est d'utiliser l'information de périodicité que l'on a sur le mot afin de ne pas parcourir tous les caractères du texte, et d'effectuer un décalage significatif à chaque étape de la recherche. Les deux algorithmes les plus connus pour la recherche exacte sont les algorithmes de Knuth-Morris-Pratt [9] (KMP) et de Boyer-Moore [37] (BM). Ces deux algorithmes ont la même complexité dans le pire des cas qui est en $O(m+n)$. La différence fondamentale entre ces deux algorithmes est que BM parcourt le mot de droite à gauche et décale le mot en fonction de ses suffixes, tandis que KMP parcourt le mot de gauche à droite, et le décalage se fait en fonction des préfixes du mot. De ce fait, tandis que KMP doit parcourir tous les caractères du texte, BM peut, dans certaines conditions,

éviter complètement de comparer certains caractères du texte. Lorsque l'alphabet Σ est suffisamment grand, l'algorithme BM est sous-linéaire en pratique. Cependant, dans le cas de petits alphabets (par exemple l'alphabet des nucléotides), la différence de performance entre les deux algorithmes est minime.

Dans le cas de la recherche de snoRNAs, nous avons choisi d'implémenter l'algorithme de Knuth-Morris-Pratt. La figure 3.1 résume le déroulement de cet algorithme.

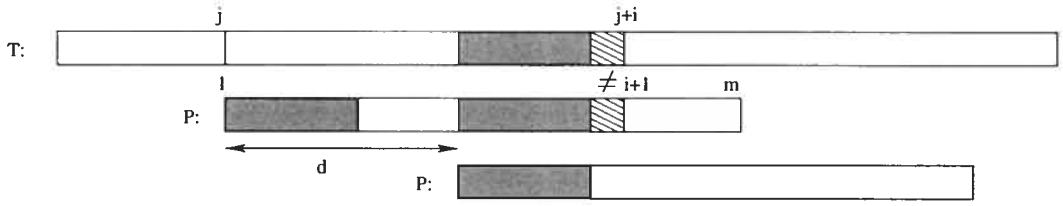


Figure 3.1 – Déroulement de l'algorithme KMP. À chaque position j de T, les caractères de P sont parcourus de gauche à droite tant qu'ils sont identiques aux caractères de T. On s'arrête au premier caractère différent, ici le caractère à la position $j+i$. Le mot est ensuite décalé en fonction de sa périodicité (décalage d calculé au cours de la phase de pré-traitement du mot P), et la recherche reprend à la position $j+d$.

3.2.2 Recherche multiple

Étant donné un texte T de taille n et un ensemble de mots $\mathscr{P} = \{P_1, \cdots, P_l\}$ de taille totale m (somme des tailles de tous les mots de $\mathscr{P}$) sur un alphabet Σ, le problème est de trouver toutes les occurrences exactes de tous les mots de $\mathscr{P}$ dans T. L'objectif est d'effectuer la recherche en un seul parcours du texte, et non pas de rechercher chaque mot séparément par un algorithme de recherche exacte.

L'algorithme de Aho-Corasick [52] (AC) est un algorithme de recherche multiple qui généralise l'approche KMP. Il est basé sur la construction d'un arbre représentant tous les préfixes des mots de $\mathscr{P}$ (voir Figure 3.2). Chaque noeud terminal représente un mot de $\mathscr{P}$, et chaque noeud interne est un préfixe d'un mot de cet ensemble. La recherche des mots de $\mathscr{P}$ dans T consiste, pour chaque position i dans T, à parcourir l'arbre AC le plus loin possible, c'est-à-dire aussi longtemps qu'il existe un caractère de l'arbre qui coïncide avec le caractère de T à la position i. De la même façon que KMP, afin

d'effectuer un décalage significatif sur le texte à l'étape suivante (ne pas redémarrer à la position $i+1$), un « pré-traitement » de l'arbre AC est nécessaire. Il consiste à déterminer une fonction d'échec : chaque noeud v de l'arbre correspondant à un préfixe p est renvoyé au noeud w correspondant au plus long suffixe des préfixes présents dans l'arbre.

L'algorithme AC a une complexité en temps $O(m+n+k)$ où k est le nombre d'occurrences des mots de $\mathscr{P}$ dans T.

$P = \{\text{abbac, ac, bacd, ababc}\}$

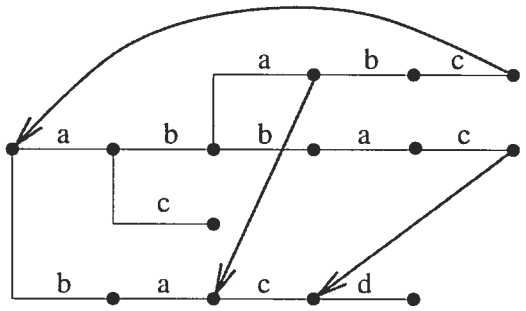


Figure 3.2 – Arbre AC pour l'ensemble $\mathscr{P} = \{abbac, ac, bacd, ababc\}$. La fonction d'échec est représentée par les flèches.

3.3 Alignement de séquences

L'alignement de séquences est un domaine classique de la bio-informatique. Elle permet de comparer des séquences biologiques en fonction d'une distance d'édition ou de similarité. C'est une partie essentielle d'une multitude de tâches en bio-informatique. Elle intervient en particulier dans la recherche de séquences homologues, l'analyse phylogénétique, l'établissement d'un consensus pour un ensemble de séquences de la même famille, ou encore l'établissement d'un consensus de structure secondaire pour des séquences d'ARN.

Différents types d'alignement de séquences sont utilisés. Un alignement global de deux séquences consiste à comparer les deux séquences sur toute leur longueur. On l'utilise, par exemple pour aligner deux séquences homologues de taille similaire. Un alignement local de deux séquences S_1 et S_2 consiste à retrouver les deux sous-séquences de S_1

et S_2 les plus similaires, et de les aligner. C'est le cas, par exemple, lorsque l'on aligne deux protéines de familles différentes, et que l'on recherche des domaines communs (sous-unités fonctionnelles conservées). Dans ce cas, il ne s'agit pas d'aligner les séquences sur toute leur longueur, mais de rechercher des parties conservées. Par ailleurs, lorsque l'on dispose d'un ensemble de séquences de gènes homologues et que l'on veut en déduire des caractéristiques communes, il est nécessaire d'effectuer un alignement multiple des séquences, c'est-à-dire de comparer toutes les séquences simultanément, et non pas les séquences deux par deux. Dans la suite, nous discutons des distances utilisées pour la comparaison de séquences, les algorithmes classiques d'alignement global et local de deux séquences, et nous introduisons brièvement le domaine de l'alignement multiple de séquences.

3.3.1 Distances entre deux séquences biologiques

Pour comparer deux séquences, on a besoin de définir une notion de distance. La distance la plus utilisée dans le cas des séquences génomiques est la distance d'édition. Elle est définie comme étant le nombre minimal d'insertions, suppressions et substitutions de caractères nécessaires pour passer d'une séquence à l'autre. Par exemple, si l'on compare les deux séquences $S_1 = CATAGTG$ et $S_2 = GTCAGGT$, le passage de S_1 à S_2 nécessite au minimum deux suppressions de caractères, deux insertions et une substitution. La distance d'édition entre ces deux séquences est donc de 5 si les coûts des opérations d'éditions sont égaux. L'alignement global correspondant est le suivant :

```
C A T - A - G T G
G - T C A G G T -
```

Le premier appariement correspond à une substitution de caractère, le deuxième à une suppression et le quatrième à une insertion. Dans la suite de ce mémoire, une substitution de caractères sera appelée «mismatch». De plus, on appellera «gap» une suite successive de suppressions ou d'insertions. Cette approche, en fonction des poids donnés aux opérations d'éditions, peut produire plusieurs alignements optimaux. Dans ce cas-ci, l'alignement constitué exclusivement de substitutions est parfaitement valide.

La distance d'édition permet de mesurer la "différence" entre deux séquences. C'est la distance la plus utilisée dans le cas des séquences nucléiques. Une mesure alternative est de considérer le degré de "similarité" entre deux séquences. Pour ce faire, on attribue un score positif dans le cas d'une identité de caractères et un score négatif dans le cas d'insertion/suppression et mismatch. Dans le cas de l'alignement de séquences protéiques, les matrices PAM et BLOSUM sont généralement utilisées pour attribuer des scores de similarité entre les paires d'acides aminés.

3.3.2 Algorithmes de programmation dynamique pour l'alignement de deux séquences

La programmation dynamique est un concept algorithmique qui consiste à résoudre un problème en commençant par résoudre tous les sous-problèmes. Elle fait appel à la récursivité. Pour ne pas calculer deux fois les mêmes sous-problèmes, les valeurs intermédiaires sont conservées dans une table.

Alignement global : Soient S et T deux séquences de tailles respectives m et n que l'on souhaite aligner par la distance d'édition. On note $D(i,j)$ la distance d'édition entre le préfixe de taille i de S et le préfixe de taille j de T. En particulier $D(m,n)$ est la distance d'édition recherchée entre S et T. La formule de récurrence suivante permet de calculer $D(i,j)$ en fonction de $D(i-1,j-1)$, $D(i,j-1)$ et $D(i-1,j)$:

$$D(i,j) = \min[D(i-1,j)+1, D(i,j-1)+1, D(i-1,j-1)+p(i,j)]$$

avec $p(i,j) = 0$ si $S_i = T_j$ et $p(i,j) = 1$ sinon. Par ailleurs il est facile de vérifier que

$$D(i,0) = i \text{ pour tout } 0 \leq i \leq m \text{ et } D(0,j) = j \text{ pour tout } 0 \leq j \leq n$$

Le calcul de chaque case $D(i,j)$ de la table de programmation dynamique D nécessite de considérer un nombre constant de cases (3 cases). Le remplissage de la table se fait donc en temps $O(mn)$. Une fois la valeur $D(m,n)$ trouvée, un alignement optimal peut-être obtenu en temps linéaire. Un exemple est donné à la figure 3.3.

		0	1	2	3	4	5
			T	A	C	G	G
0		0	1	2	3	4	5
1	T	1	0	1	2	3	4
2	C	2	1	1	1	2	3
3	G	3	2	2	2	1	2
4	C	4	3	3	2	2	2

T – C G C
T A C G G

Figure 3.3 – Alignement des deux séquences $\mathscr{S} = TCGC$ et $\mathscr{T} = TACGG$. La distance d'édition entre les deux séquences est 2 (dernière case en bas à droite). Les flèches permettent de retracer un alignement entre les deux séquences. L'alignement correspondant est indiqué à droite de la table de programmation dynamique.

Si on considère une valeur de similarité plutôt qu'une distance, le min de la formule de récurrence est remplacée par un max.

Alignement local : Dans le cas de l'alignement local on considère une distance de similarité plutôt que la distance d'édition. L'algorithme que nous décrivons ici est connu sous le nom de Smith-Waterman [50].

Étant donné une séquence S de taille m et une séquence T de taille n, le problème est de trouver deux sous-séquences de S et T de similarité maximale (voir figure 3.4).

C A G C A C [T T – G G A T] T C T C G G
T A G T [T T A G G – T] G G C A T

Figure 3.4 – Alignement local de deux séquences pour la distance de similarité suivante : 2 pour une paire identique de nucléotides, et -1 pour une insertion/suppression ou mismatch. Les sous-séquences alignées sont celles entre crochets. La valeur de similarité est de 8.

Soit $V(i,j)$ la valeur maximale de similarité locale entre le préfixe de taille i de S et le préfixe de taille j de T. Les relations de récurrence pour le calcul de $V(i,j)$ sont les suivantes :

$$V(i,0) = 0,\ \ V(0,j) = 0 \text{ pour tout } i,\ j$$

$$V(i,j) = max[0, V(i-1,j-1) + f(s_i,t_j), V(i-1,j) + f(s_i,-), V(i,j-1) + f(-,t_j)]$$

où f est la fonction qui attribue un score de similarité à chaque appariement.

La différence principale avec l'alignement global est la ré-initialisation de la récurrence à 0 dès que V atteint une valeur négative. Ceci permet d'ignorer un certain nombre de caractères au début des séquences.

Afin de retrouver la valeur maximale de similarité entre les séquences S et T il suffit de trouver une case de la table de programmation dynamique ayant une valeur maximale.

3.3.3 Recherche approchée de mots

Supposons que l'on possède une séquence codante S de taille m pour un gène g et que l'on recherche les homologues de ce gène dans une base de données biologique T de taille n. Dans ce cas, la recherche exacte n'est pas suffisante étant donné qu'un nombre significatif de mutations peut avoir affecté la famille de gènes. Il faut alors rechercher la séquence de façon "approchée", c'est-à-dire en autorisant un certain nombre d'insertions, suppressions et substitutions de caractères. Le problème se ramène alors à celui de l'alignement de la séquence S avec la base de donnée T, sachant que l'on veut aligner le mot S en entier avec une sous-séquence de T. Dans ce cas les relations de récurrence sont les suivantes :

$$D(i,0) = i, D(0,j) = 0 \textit{ pour tout } i,j$$

$$D(i,j) = \min[D(i-1,j)+1, D(i,j-1)+1, D(i-1,j-1)+p(i,j)]$$

avec $p(i,j) = 0$ si $S_i = T_j$ et $p(i,j) = 1$ sinon.

Supposons que l'on recherche toutes les positions de S dans T présentant au plus k erreurs (insertions, suppressions, mismatchs). Il suffit alors de retrouver toutes les positions j dans le texte telles que la case $D(m,j) \leq k$.

Cet algorithme de programmation dynamique pour la recherche approchée de mots

se fait en temps $O(mn + rm)$, où r est le nombre d'occurrences de S dans T. Cet algorithme quadratique devient rapidement inefficace sur des banques de données biologiques, de taille généralement considérable (millions de nucléotides pour un seul génome). Il est donc nécessaire d'accélérer la recherche approchée et d'obtenir des algorithmes sous-linéaires en moyenne, quitte à perdre en précision. C'est la raison pour laquelle les heuristiques de type FastA et Blast ont été développées. Le mot « heuristique » signifie que l'algorithme développé n'est pas garanti d'obtenir la meilleure solution. Dans le cas d'heuristiques pour la recherche approchée, certaines des occurrences trouvées peuvent présenter un taux d'erreur supérieur à celui demandé. La performance des heuristiques est généralement vérifiée empiriquement. Nous présentons plus en détail l'algorithme BLAST [11], l'algorithme le plus utilisé pour la recherche dans les banques de données biologiques.

3.3.4 BLAST

L'idée générale de BLAST, ainsi que de la plupart des méthodes de filtrage utilisées pour la recherche d'un mot dans un texte, est d'effectuer un premier parcours du texte afin d'éliminer toutes les parties qui ne sont pas susceptibles de contenir une occurrence du mot. Les occurrences potentielles restantes sont ensuite validées ou éliminées en utilisant une méthode de recherche approchée, puis une méthode de pointage, permettant de filtrer les occurrences selon leurs scores. Afin que l'algorithme élaboré soit rapide, il faut qu'il y ait un bon compromis entre la phase de recherche et la phase d'élongation des candidats.

Plus précisément, l'algorithme BLAST peut être divisé en trois étapes :

1. L'algorithme construit la liste de tous les sous-mots (graines) de taille w (12 est la valeur par défaut) de la séquence S recherchée, donc $m - w - 1$ mots où m est la taille de S.
2. La phase de recherche consiste à retrouver toutes les occurrences de ces sous-mots dans le texte. Un « High Scoring Pair » (ou HSP) est une occurrence qui contient deux sous-mots distincts de S, notés w_1 et w_2, sur T.

Figure 3.5 – Fragmentation du mot requête de taille m en ses $m - 11$ sous-mots de taille 12.

3. Pour chaque HSP, on passe en phase d'extension des deux côtés des sous-mots w_1 et w_2. Cette phase est terminée lorsque le score d'alignement tombe en bas d'une certaine valeur. Cet alignement est ensuite accepté si son score est supérieur à un seuil donné S, donc considéré comme un « hit » ; une position dans la base de donnée qui possède une séquence similaire. Cette étape est arrêtée lorsqu'une extension maximale du HSP est atteinte. Un exemple de cette procédure est décrit à la figure 3.6. La figure montre d'abord comment deux graines de tailles 12 se positionnent sur la base de données, puis, après allongement des graines, on obtient un hit, un candidat qui n'a pas exactement la même séquence (voir le trou entre les alignements).

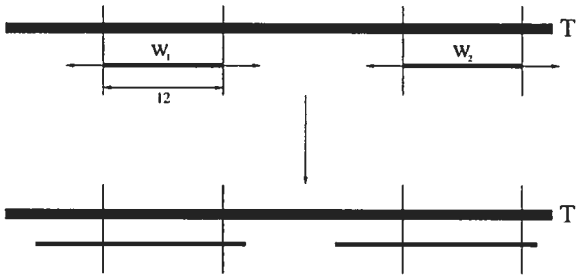


Figure 3.6 – Exemple d'un HSP et de son élongation.

La procédure est rapide parce que l'on utilise dans la première phase la recherche exacte d'un nombre restreint de petites séquences, et que la probabilité d'obtenir des paires de graines qui vont pouvoir s'étendre suffisamment pour être acceptées est encore plus petite. La complexité de la deuxième et de la troisième phase dépend de la taille des mots ; dans l'exemple, la taille a été fixée à la valeur par défaut, 12, mais on peut très bien spécifier une autre valeur. La complexité dépend aussi du seuil d'acceptation du HSP (lui aussi pouvant être modifié à l'exécution) ; des mots de grande taille réduisent le nombre

de hits à étendre, diminuant donc le temps pour la troisième phase, mais introduisent une perte de sensibilité lors de la première phase de recherche. Le programme a été utilisé avec les paramètres par défaut, soit la version 2.2.10.

Le programme sort les hits qui sont inférieurs à un score donné après l'élongation à la troisième étape. Ce score est exprimé soit en P-value, soit en E-value. Le score E-value correspond à la probabilité d'obtenir un HSP sur l'ensemble de la banque de données ; plus cette valeur est faible, plus le HSP obtenu ressemble à la séquence requête. La P-value est définie comme 1 - la chance d'obtenir 0 HSP de score $\geq S$. La E-value et la P-value deviennent identiques lorsque la E-value ≤ 0.01.

BLAST est une heuristique à cause de la définition même de la graine ; on peut imaginer une situation où le HSP idéal serait composé de graines d'une taille inférieure à celle utilisée par le programme, et, étant donné que ces mots précis ne seraient pas dans la liste de recherche, cette situation ne serait pas retrouvée par BLAST. Cette possibilité ne nous préoccupe guère car, comme nous le verrons dans la section 5.2.4, le seuil de E-value utilisé pour filtrer mes occurrences rend cette situation très improbable.

3.3.5 Alignement multiple

Supposons que l'on dispose d'un ensemble de séquences appartenant à la même famille, c'est-à-dire apparenté par la structure, la fonction ou des relations d'évolution communes. Dans notre cas les séquences considérées seront les gènes codants les snoRNAs de type C/D, et les sites de reconnaissance de l'enzyme RNT1P. Afin de déterminer un consensus pour la famille, autrement dit les caractéristiques communes à toutes les séquences de la famille, on doit effectuer un alignement de toutes les séquences simultanément. L'alignement multiple obtenu permet alors de déduire une représentation de la famille. Nous décrivons ci-dessous les représentations les plus utilisées :

- Une *séquence consensus* ou signature, est une séquence contenant, à chaque position, le caractère le plus fréquent à cette position de l'alignement.
- Une *matrice consensus* est une matrice contenant la fréquence d'apparition de chaque caractère (nucléotide ou acide aminé) à chaque position dans l'alignement (voir table 3.1).

Un exemple d'alignement et de ses représentations est donné à la figure 3.7. La table 3.1 nous donne la matrice consensus correspondante.

$$
\begin{array}{c}
\begin{matrix} C_1 & C_2 & C_3 & C_4 & C_5 & C_6 & C_7 \end{matrix} \\
\begin{bmatrix}
A & A & G & A & A & - & A \\
A & T & - & A & A & T & G \\
C & C & G & - & G & - & G \\
C & C & - & A & G & T & T \\
C & C & G & - & G & - & -
\end{bmatrix} \quad \text{Alignement} \\
\\
C \; C \; G \; A \; G \; - \; G \quad \text{Signature}
\end{array}
$$

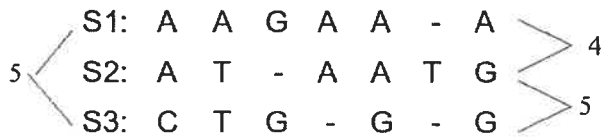
Figure 3.7 – Exemple d'un alignement multiple de cinq séquences d'ADN et le consensus produit (signature).

Pour aligner de façon « optimale » un ensemble de séquences, on a besoin d'une mesure de similarité ou de distance. L'objectif est alors d'obtenir l'alignement multiple qui maximise le score de similarité ou qui minimise la distance. Étant donné un alignement multiple $\mathscr{A}$, le score « Sum-of-Pairs » (SP) de $\mathscr{A}$ est la somme de tous les scores des alignements induits pour toutes les paires de séquences (voir figure 3.8). On voit dans la figure que la distance entre les séquences $S1$ et $S3$ est de 5 puisque 5 colonnes sur 7 n'affichent pas le même caractère pour ces deux séquences. Dans cet exemple, le score SP est de 14, on peut en conclure que les 3 séquences ont peu de similarité entres elles.

L'alignement multiple de séquences fait partie des domaines de la bio-informatique les plus actifs. Une multitude d'algorithmes ont été élaborés, et de nouvelles avenues continuent à être explorées. Un "bon" alignement multiple devrait refléter les mutations survenues au cours de l'évolution qui ont donné lieu aux séquences observées.

	C_1	C_2	C_3	C_4	C_5	C_6	C_7
A	0.40	0.17	0.00	0.60	0.40	0.00	0.17
C	0.60	0.60	0.00	0.00	0.00	0.00	0.00
G	0.00	0.00	0.60	0.00	0.60	0.00	0.40
T	0.00	0.17	0.00	0.00	0.00	0.40	0.17
-	0.00	0.00	0.40	0.40	0.00	0.60	0.17

Tableau 3.1 – Matrice consensus pour l'alignement présenté à la figure 3.7.



Score SP = 14

Figure 3.8 – Exemple de scores induits par la méthode SP. Le score est exprimé en distance d'édition.

D'un point de vue algorithmique, le problème de trouver un alignement multiple qui minimise le score SP a été prouvé NP-complet (aucun algorithme polynomial ne peut être trouvé). C'est la raison pour laquelle les algorithmes développés dans ce cas sont des heuristiques.

Dans le cadre de ma recherche, l'alignement multiple utilisé consiste à ancrer les séquences autour d'un motif commun et d'aligner le reste des séquences de la meilleure façon possible. Ceci est efficace puisque les séquences comparées ne possèdent pas beaucoup d'insertions, de suppressions et de mismatchs entre elles.

3.4 Conclusion

Les méthodes et algorithmes présentés dans ce chapitre sont couramment employées dans le domaine de la bio-informatique. Ils sont à la base de nombreuses applications présentées dans le chapitre suivant.

CHAPITRE 4

MÉTHODES EXISTANTES POUR LA RECHERCHE D'ARN

4.1 Introduction

Il existe plusieurs logiciels permettant de rechercher divers types d'ARN, ou tout autre type d'information contenue dans l'ADN. La recherche de familles d'ARN est basée sur diverses caractéristiques liées à la structure primaire (séquence), secondaire ou tertiaire de la molécule.

Les méthodes de recherche de structures d'ARN peuvent être classées en deux catégories : les méthodes de recherche spécifique ne permettant de rechercher qu'un type d'ARN, et dont on ne peut, généralement, modifier l'information de recherche (« tailor-made algorithms » : méthodes sur mesure), et les méthodes de recherche généralistes prenant en entrée une description de la molécule à rechercher, et permettant de retrouver, dans un génome ou une base de données, tous les motifs se conformant à cette description [16, 23, 31, 48]. Plusieurs algorithmes de recherche "sur mesure" ont été développés pour les structures d'ARN les plus étudiées et les mieux caractérisées. C'est le cas, en particulier, des structures d'ARN de tranfert [42, 45], et des introns de type I et II [28] et des snoRNAs [15, 51]. Cependant, la majorité des familles d'ARN sont très difficiles à caractériser, et aucun programme de recherche spécifique n'existe pour ces familles. Les méthodes générales sont alors utilisées pour rechercher diverses parties conservées des structures d'ARN. Ces méthodes permettent d'essayer diverses caractéristiques, et contribuent à affiner la connaissance de ces structures.

Une autre classe de méthodes de recherche existe : il s'agit des algorithmes basés sur l'entraînement. Cette approche, appartenant au domaine de l'apprentissage machine, consiste à laisser au logiciel le soin de trouver des caractéristiques communes à un ensemble de séquences données en paramètres (généralement de la même famille), et ensuite de tester le modèle créé en le recherchant dans une base de données.

Ce chapitre a pour but d'introduire plusieurs méthodes liées à la recherche de struc-

tures d'ARN, et d'expliquer dans quelle mesure ma méthode tire profit de celles-ci, ou, inversement, pour quelles raisons elles sont inappropriées. Étant donné que ce mémoire s'intéresse à la recherche de snoRNAs, j'introduirai à la section 4.5 les approches "sur mesure" existantes dédiées à la recherche de snoRNAs de type C/D et de type H/ACA. Puisque la plupart des approches "sur mesure" utilisent comme sous-modules des algorithmes de recherche généralistes (pour rechercher des sous-parties conservées de la molécule), nous commencerons par introduire quelques méthodes généralistes à la section 4.4.

Afin de pouvoir développer une méthode sur mesure pour la recherche de structures secondaires, on a besoin d'une caractérisation de la famille recherchée. La méthode naturelle consiste à effectuer un alignement multiple des séquences connues appartenant à la famille, et d'en tirer une représentation sous forme de séquence consensus, matrice consensus ou d'expression régulière (voir chapitre 3, section 3.3.5). D'autres méthodes existent pour la prédiction d'un consensus de structure secondaire. Nous commençons donc ce chapitre en introduisant deux méthodes reliées à la prédiction de structures secondaire : la méthode classique de repliement par minimisation de l'énergie libre , ainsi qu'une méthode basée sur l'entraînement.

4.2 Prédiction de structures secondaires

4.2.1 Minimisation de l'énergie libre

Lorsque l'on prédit la structure secondaire d'une molécule d'ARN, nous n'avons en entrée que la séquence et nous souhaitons connaître le repliement dans l'espace. Comme il a été vu en section 2.4, certaines séquences adoptent des structures compliquées, l'ARNt par exemple. La prédiction de structure secondaire repose sur trois grands principes :

- La structure originale sera la structure la plus stable.
- L'énergie associée à chaque position est influencée de façon locale seulement par la séquence et la structure.
- La structure ne forme pas de pseudonoeuds.

Les deux logiciels décrits dans cette section, MFold [40] et Vienna RNA Package [20], se basent sur ce principe afin de trouver la structure optimale, c'est-à-dire celle que nous devrions retrouver *in vivo*.

4.2.2 MFold

L'algorithme utilise une méthode de programmation dynamique : pour calculer l'énergie minimale de repliement d'une séquence S, on calcule l'énergie minimale de repliement de chaque sous-séquence de S. Une énergie négative signifie que la structure est stable, alors qu'une énergie positive signifie qu'elle est instable. On calcule l'énergie libre d'une structure (Δ G) comme étant la somme des contributions individuelles plusieurs sous-structures : les boucles, les appariements de bases, les éléments de structures secondaires. Les paramètres énergétiques reliés à chacune de ces sous-structures (voir figure 2.2) sont dérivés d'expériences de thermodynamique.

Les critiques à propos de ce logiciel sont multiples [39]. On peut tout d'abord souligner que les conditions chimiques *in vivo* peuvent être différentes des expériences de thermodynamique ayant permis de trouver les contributions des sous-structures à l'énergie globale. La structure tertiaire n'est pas prise en compte, et l'interaction avec des molécules retrouvées dans le cytoplasme ne l'est pas non plus. Malgré ce qu'en disent ses détracteurs, mFold reste un logiciel classique et le plus utilisé pour la prédiction de structure secondaire.

4.2.3 Vienna RNA Package

Ce logiciel, bien qu'une partie soit presque identique à MFold (procédure RNAfold), est une collection de programmes spécialement orientés vers le traitement de l'information ARN. RNAsubopt permet d'énumérer toutes les structures sous-optimales du point de vue énergétique. Un autre programme calcule le repliement inverse (RNAinverse), qui permet de calculer l'ensemble de séquences ayant une structure secondaire similaire à celle donnée en entrée. Le logiciel vient aussi avec une panoplie de fonction pour évaluer la stabilité d'une structure.

4.3 Méthodes basées sur l'entraînement

4.3.1 Cove

Cove [44] est une méthode probabiliste permettant de trouver des caractéristiques communes à un ensemble de séquences. La méthode développée et présentée à la figure 4.1 est abordée ici dans ses grandes lignes :

1. À l'aide d'un algorithme d'alignement multiple de séquences, aligner l'ensemble des séquences.
2. Appliquer la statistique d'information mutuelle (MI, voir section 5.2.2) sur toutes les possibilités de couples de colonnes de la matrice consensus afin de trouver des règles de co-variation.
3. Utiliser les règles de co-variation pour établir un modèle de co-présence (structure secondaire consensus).
4. Repartir à l'étape 1 avec un nouvel alignement de séquences, celui défini par la structure consensus, et itérer les trois premières étapes jusqu'à ce qu'un seuil de satisfaction soit atteint. Cette boucle est gérée par un algorithme EM [33] (expectation maximisation).

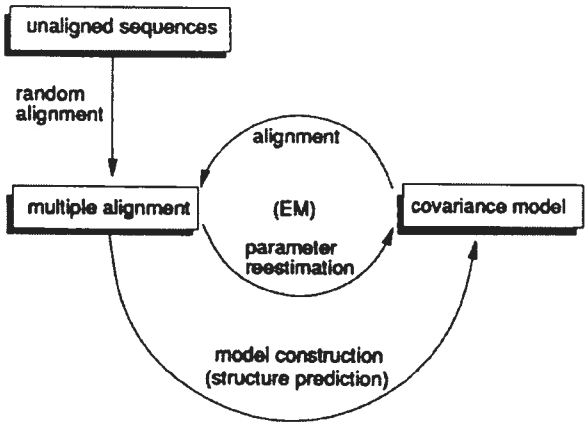


Figure 4.1 – Procédure du programme Cove (figure 3 tirée de [44]).

La statistique MI permet de mettre en relation les bases qui sont co-présentes (appariements, nucléotides essentiels, etc). On qualifie ce programme de probabiliste parce que le modèle de covariance créé est dérivé d'un modèle de Markov caché [2], et un score est associé à nos prédictions. Le programme, en plus de fournir de l'information sur la structure primaire et secondaire de l'ensemble de séquences données en entrée, permet aussi de tester de nouvelles séquences sur le modèle ainsi créé, et donc d'obtenir une mesure de distance entre l'ensemble des séquences d'entraînement et un nouvel ensemble de séquences, distance traduite sous forme de score de probabilité. Dans le cas où la structure secondaire ne change pas au cours du temps, par exemple dans une famille fonctionnelle de molécules, la statistique MI permet une approximation acceptable du consensus de la famille.

Cove a été utilisé comme étape de filtrage dans la procédure globale de recherche présentée au chapitre 5. On utilisait le signal de reconnaissance à l'enzyme RNT1P comme séquence à analyser. Une analyse de cette composante dans notre modèle de recherche nous a permis de constater que cette étape n'était pas assez sensible par rapport aux données d'entraînement du modèle. En effet, pour rendre cette étape utile (éliminer beaucoup de candidats FP), il était nécessaire de spécifier un seuil élevé faisant en sorte qu'on perdait beaucoup de TP. Cette étape a été mise de côté mais la statistique MI a été jugée pertinente dans notre recherche (voir section 5.2.2).

4.4 Méthode généraliste

Les méthodes généralistes sont utilisées lorsque l'utilisateur veut avoir un contrôle accru sur le type d'ARN recherché. Elles restent la seule option lorsqu'aucun algorithme de recherche spécifique n'existe. Ces méthodes prennent en entrée une description de la molécule à rechercher et un génome (ou une base de données), et retrouvent les sous-séquences du génome qui vérifient ces caractéristiques. Cette description est souvent une combinaison de critères de recherche pour la structure primaire et secondaire, et parfois tertiaire. Le principal désavantage est que, bien qu'on n'ait moins besoin de connaître la structure de la molécule qu'avec des méthodes spécifiques, il faut tout de même fournir

de l'information.

Plusieurs méthodes existent dans cette approche : Biosmatch [16], RNAMST [48], l'approche de Pavesi [31] et RNAMOTIF [23] en sont des exemples. Je discuterai de RNAMOTIF car cette approche est la plus utilisée par les biologistes. Elle permet une recherche rapide d'un descripteur composé d'une combinaison de conservations de structures primaires et secondaires.

4.4.1 RNAMOTIF

L'algorithme de recherche RNAMOTIF [23] utilise une méthode de backtracking permettant de rechercher de façon séquentielle les différentes parties d'un descripteur fourni en entrée. Deux exemples de descripteurs de RNAMOTIF sont donnés à la figure 4.2.

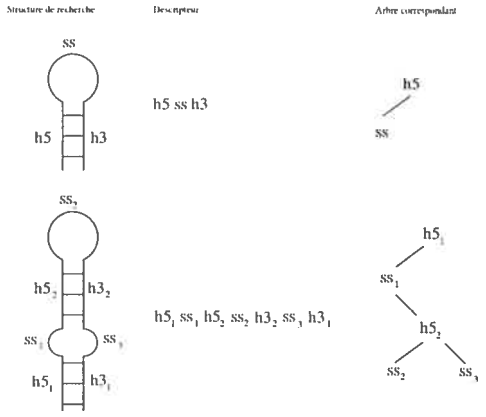


Figure 4.2 – Structure recherchée, et interprétation de la description par le programme RNAMOTIF. H représente une portion hélicale, avec 5 (respect. 3) représentant le brin 5' (respect. 3'), ss représente une structure simple brin.

Comme le montre la troisième section de la figure 4.2, le programme transforme l'information contenue dans le descripteur en arbre binaire de recherche défini comme suit :

- La racine est un élément d'hélice (partie appariée).
- Le fils gauche est un élément interne d'une hélice (une boucle si ss, une autre expression dans l'alphabet du descripteur sinon).

– Le fils droit est considéré comme une sous-structure secondaire.

La contrainte supplémentaire est qu'un élément simple brin (ss) ne peut avoir de fils gauche, mais peut avoir un fils droit. Cette contrainte permet de considérer des motifs plus complexes comme deux hélices reliées par un brin charnière.

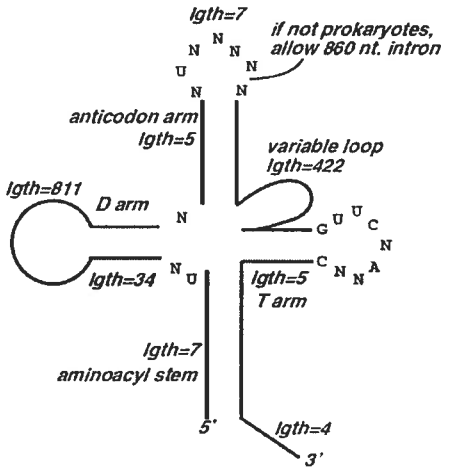
En plus de construire l'arbre de recherche, le programme vérifie s'il n'y a pas d'ambiguïtés dans le descripteur (chevauchement des différentes expressions).

La procédure d'exploration de l'arbre de recherche est faite selon l'algorithme DFS (« depth first search », ou recherche en profondeur). Ceci permet de cibler en premier les régions terminales des tiges-boucles, et est un algorithme récursif simple et efficace. Lorsqu'un candidat répond aux exigences du descripteur, une fonction du programme permet de lui attribuer un score. Le score est évalué en fonction d'un ensemble de règles définies dans le descripteur, reliés à des paramètres d'énergie libre [40], et en fonction d'une prépondérance de composition de certains nucléotides qui forcent la séquence dans une structure particulière. Cette dernière est utile lorsque l'on sait qu'une expression pourrait favoriser des candidats qui se retrouvent dans des régions A-U riches, et que l'on ne veut pas de ces candidats.

Un descripteur plus complet pour l'ARNt (voir figure 2.1) est présenté en figure 4.3. Voici les différentes étapes de ce descripteur :

1. On commence par spécifier que les wobbles sont considérés comme un appariement WC.
2. H1 : on recherche l'hélice « aminoacyl », taille 7, 1 substitution au maximum.
3. S1 : on recherche la section interne de la boucle multiple (entre H1 et H2), taille 2.
4. H2 : on recherche l'hélice « D », taille $\in [3,4]$, 1 substitution au maximum.
5. S2 : on recherche la boucle de l'hélice H2, taille $\in [8,11]$.
6. S3 : on recherche la région interne de la boucle multiple (entre H2 et H3), taille 1.
7. H3 : on recherche l'hélice « anticodon », taille 5, 1 substitution au maximum.
8. S4 : on recherche la boucle de l'hélice H3, taille 7.

9. S5 : on recherche la boucle variable, taille $\in [4,22]$.
10. H4 : on recherche l'hélice « T », taille 5, 1 substitution au maximum.
11. S6 : on recherche la boucle de l'hélice H4, taille 7.
12. S7 : finalement, du côté 3' de l'hélice H1, on cherche un brin de taille 4.
13. Pour le score, on vérifie pour 8nt la nature de ceux-ci. La molécule est rejetée si le nucléotide en question n'est pas celui voulu.
14. Finalement, la structure est évaluée avec la fonction mesurant l'énergie libre.




```
parms
  wc += gu;

descr
     h5(tag='h1',len=7,mispair=1,ends='mm')
        ss(tag='s1',len=2)
        h5(tag='h2',minlen=3,maxlen=4,mispair=1,ends='mm')
           ss(tag='s2',minlen=8,maxlen=11)
        h3(tag='h2')
        ss(tag='s3',len=1)
        h5(tag='h3',len=5,mispair=1,ends='mm')
           ss(tag='s4',len=7)
        h3(tag='h3')
        ss(tag='s5',minlen=4,maxlen=22)
        h5(tag='h4',len=5,mispair=1,ends='mm')
           ss(tag='s6',len=7)
        h3(tag='h4')
     h3(tag='h1')
     ss(tag='s7',len=4)

score
{
  n = 0;
  if (ss['s1',1,1] != "u")  n++;
  if (ss['s4',2,1] != "u")  n++;
  if (h5['h4',5,1] != "g")  n++;
  if (ss['s6',1,1] != "u")  n++;
  if (ss['s6',2,1] != "u")  n++;
  if (ss['s6',3,1] != "c")  n++;
  if (ss['s6',5,1] != "a")  n++;
  if (h3['h4',1,1] != "c")  n++;

  if (n > 1) REJECT;

  SCORE = efn( h5['h1'],ss['s7'] );
}
```

Figure 4.3 – Descripteur plus complexe pour RNAMOTIF permettant de retrouver un ARN de transfert (figure tirée de l'article [7].

On voit donc, par l'analyse du descripteur donné en figure 4.3, que les caractéristiques peuvent devenir très concises tout en permettant d'avoir des régions avec une certaine flexibilité.

4.5 Méthodes sur mesure

Des méthodes de recherches spécifiques ont été appliquées avec succès pour retrouver des ARNt : FAStRNA [42], tRNAscan-SE [45], des introns de groupes 1 et 2 : CI-

TRON [28], de même que des snoRNAs : snoSCAN [51], FISHER [15]. Dans tous les cas, le programme ne permet de retrouver que la classe d'ARN cible. Étant donné que mon sujet de recherche porte sur les snoRNAs, j'aborderai les méthodes sur mesure concernant les snoRNAs de type boîte C/D, et ensuite celles concernant les snoRNAs de type H/ACA.

4.5.1 Recherche de snoRNAs de type C/D

Le programme de recherche de snoRNAs de type C/D de S.Eddy et T.Lowe s'intitule snoSCAN [51]. Le modèle de recherche développé est un modèle probabiliste composé de neuf états présentés à la figure 4.4. Ces états sont dérivés de six caractéristiques de structure primaire et secondaire des snoRNAs :

- Boîte C (consensus AUGAUGA).
- Boîte D (consensus CUGA).
- Région variable entre la boîte C et la séquence guide (ASE).
- Séquence guide (ASE).
- Boîte D'.
- Hélice terminale.

Les caractéristiques du modèle sont présentés à la figures 4.4. Le modèle utilisé pour représenter la recherche d'un snoRNA est un modèle de Markov caché (HMM) [2]. Subséquemment, chaque état du modèle décrit une caractéristique. Par exemple, la boîte C est représentée par un autre HMM sur 7 bases. Un HMM doit être entraîné avant de faire une recherche ; les auteurs ont entraîné leur modèle avec un nombre relativement réduit d'exemples (35 snoRNAs de type C/D humains).

Lorsqu'une séquence requête passe au travers du modèle, chacun des états, et le chemin emprunté, contribuent au score final qu'aura ce candidat. Le problème avec la méthode proposée en figure 4.4 est qu'elle associe un poids extrêmement élevé à la séquence guide : la contribution relative au score final des étapes 4 et 9 sont dans les trois les plus importantes (avec la recherche de la boîte C). Ces états du modèle sont régis par les connaissances actuelles que nous avons des sites de méthylation. À l'époque de la sortie de l'article, ils avaient 42 sites de méthylation non associés à des snoRNAs sur

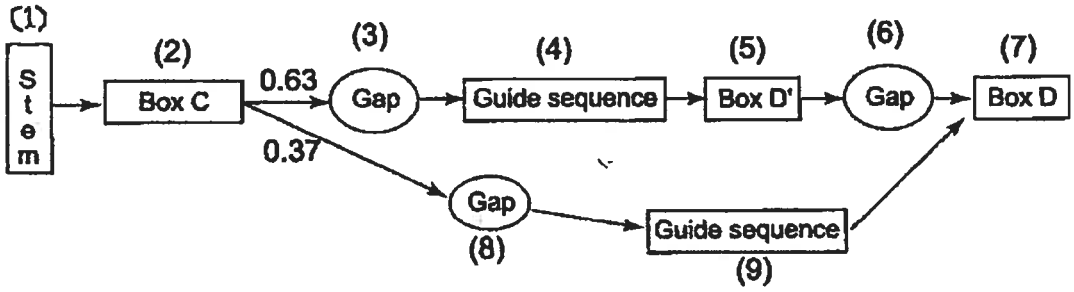


Figure 4.4 – Modèle probabiliste développé par Lowe et Eddy (figure 2 de [51]).

l'ARNr, et la recherche a été orientée autour de 42 possibilités. De ce fait, le modèle ne retient que les snoRNAs qui agissent en tant que guide de méthylation sur l'ARNr, et exclut ceux qui agissent de concert avec le spliceosome (snRNA), ou qui ont une fonction inconnue.

Un des critères utilisés par l'algorithme pour filtrer les candidats potentiels obtenus consiste à vérifier si ceux-ci se retrouvent dans des ORF, et les éliminer le cas échéant. Cette optique de recherche est intéressante, mais étant donné que 6 snoRNAs répertoriés sont retrouvés dans des introns de protéines, il est difficile de justifier cette approche.

4.5.2 Recherche de snoRNAs de type H/ACA

Bien que les snoRNAs de type H/ACA soient différents des snoRNAs de type C/D, les approches générales développées pour la recherche des deux types de snoRNAs sont reliées, et il est utile de connaître ces méthodes avant de se lancer dans l'élaboration d'une nouvelle méthode de recherche de snoRNA.

Les snoRNAs de type H/ACA se caractérisent par 2 motifs conservés ou boîtes : la boîte H (consensus $AN_1AN_2N_3N_4N_5$) et la boîte ACA (consensus ACA). Deux autres motifs d'importance, soit ψ_3 et ψ_4, permettent au snoRNA de reconnaître l'endroit de pseudouridylation sur l'ARN cible. Ce sont les séquences complémentaires de la zone dans laquelle le nucléotide à être modifié se trouve. Cette famille semble adopter une structure secondaire plus complexe que pour le cas des snoRNAs de type C/D comme le montre la figure 4.5.

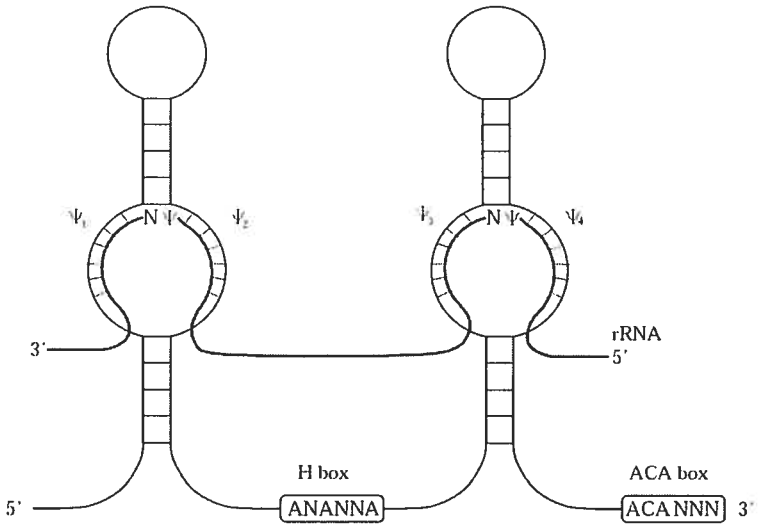


Figure 4.5 – Description schématique d'un snoRNA de type H/ACA. On remarque les 2 boîtes conservées(H et ACA) ainsi que les séquences guide ψ_3 et ψ_4, qui se retrouvent complémentaires au brin d'ARN ribosomal (figure 1 de [15]).

Plusieurs groupes de recherche se sont intéressés aux snoRNAs de type H/ACA, et plusieurs méthodes de recherche de tels gènes existent. La première [15], était pionnière dans la recherche spécifique reliée à cette molécule. L'autre [17], plus récente, se veut une méthode plus sensible.

Approche de Edvardsson *et al.* [15] Cette approche, similaire à celle présentée en section 4.5.1 dans le sens où l'on recherche plusieurs caractéristiques de la séquence primaire et que l'on accorde un score à chacune d'entre elles, est toutefois différente car elle accorde beaucoup plus d'importance à la structure secondaire de la molécule. Les détails de la procédures sont indiqués ici :

1. Sur la base de données et à l'aide d'un modèle probabiliste, rechercher la boîte H.
2. Rechercher les sites de pseudouridylation ψ_3 et ψ_4.
3. Rechercher la boîte ACA à 14nt max du site ψ_4.
4. Filtrer les occurrences par structure secondaire.

Un modèle probabiliste est appliqué sur le consensus de la boîte H ($AN_1AN_2N_3N_4N_5$) et concerne les 5 positions N_i où N_i signifie n'importe quel nucléotide dans $\{A, C, G, U\}$. Ce modèle associe des probabilités pour chaque position et chaque nucléotide. Les occurrences peuvent ensuite être filtrées selon leur score de probabilité. Les scores sont déduits de l'observation de 16 snoRNAs connus.

Les séquences ψ_3 et ψ_4 sont similaires au motif ASE (voir section 4.5.1) pour les snoRNAs de type C/D, la recherche de ces motifs est donc fonction des connaissances actuelles des sites de pseudouridylation sur l'ARNr. Voici les différents critères qu'un candidat doit satisfaire pour l'étape 2 de la recherche :

- Le candidat doit pouvoir avoir une séquence complémentaire de 3 à 10 paires de bases avec au plus un wobble.
- La somme des longueurs des deux motifs doit être au moins de 8.

La dernière étape consiste à passer les candidats au travers de plusieurs filtres de structure secondaire à l'aide de mFold [40] et Vienna RNA package [20]. On attribue ensuite un score aux snoRNAs relativement à la qualité et la stabilité de la structure secondaire.

La méthode proposée se basait sur un ensemble restreint de snoRNAs connus : 13 seulement étaient connus et ont été utilisés pour paramétrer le modèle. Cette approche a permis d'obtenir finalement 50 candidats dont trois se sont avérés exacts ; on doit par contre souligner que la méthode ne générait aucun FN.

Approche de Eo *et al.* [17] Cette approche est similaire à celles de T.Lowe [51] et Edvardsson [15] dans le sens où elle reprend le principe de modèle de Markov caché pour détecter la boîte H. Elle reprend aussi des éléments de l'approche d'Edvardsson [15] en ce qui a trait au repliement d'énergie minimale. Le modèle est composé de trois filtres. Le premier filtre consiste en la recherche des boîtes H et ACA, le modèle de recherche de la boîte H étant un HMM entraîné sur l'ensemble de snoRNAs connus. Les contraintes supplémentaires sont relatives à la distance entre les 2 types de motifs.

Le filtre suivant, HIT (« Homeomorphically Irreducible Trees »), a été utilisé pour prédire la structure secondaire de la molécule. Les auteurs utilisent finalement les para-

mètres énergétiques comme troisième filtre pour séparer les structures bien déterminées des autres. Ils terminent la recherche par une analyse *de visu* des différents candidats, par génomique comparative (voir section 5.2.4), et en regardant si le candidat peut s'apparier avec une région connue de pseudouridylation. Leur méthode retrouve 12 vrais snoRNAs de type H/ACA sur 50.

Une caractéristique qui sépare cette méthode des deux autres méthodes spécifiques vues jusqu'à maintenant est qu'au lieu de faire une recherche sur le génome entier, ils utilisent une base de données contenant 30 séquences génomiques de protéines nucléolaires. Ces séquences ont été choisies parce que ces protéines sont reliées soit au ribosome, soit au complexe protéique qui est responsable de la pseudouridylation. Ceci est une façon de contourner la grande quantité de faux positifs qui serait générée par une recherche génomique, et est probablement la meilleure façon lorsque l'on effectue une recherche sur le génome humain.

4.5.3 Recherche de sites de clivage à l'enzyme RNT1P

L'équipe de Sherif Abou Elela à Sherbrooke a développé une méthode combinant les approches *in vitro* et *in silico* afin de trouver des sites de clivage à l'enzyme RNT1P près de snoRNAs connus. La méthode s'intitule DRSD (« Dynamic RNT1P Substrate Database »), et est décrite à la figure 4.6.

Cette méthode prend en entrée un ensemble de snoRNAs (peu importe la famille), avec les séquences génomiques en amont et en aval de ces candidats, et un ensemble connu de substrats à l'enzyme RNT1P. Le consensus AGNN et les trois appariements de type WC proximaux qui caractérisent la « tetraloop » sont recherchés à l'aide du logiciel RNAMOTIF [7]. La banque de candidats ainsi obtenue, ainsi que l'ensemble des substrats connus deviennent les entrées du logiciel Vienna RNA package [20], qu'on utilise afin de trouver la structure secondaire. Cette étape est effectuée soit en forçant la structure AGNN tetraloop, soit en la laissant libre. La structure est ensuite quantifiée avec différentes approches et un score s_1 en est déduit, $s_1 \in [0, 1]$. Ce score a une valeur proche de 1 lorsque l'hélice ne contient que peu de bulges et de boucles internes. Dans un même temps, les candidats tetraloop et les substrats connus sont évalués en regard de

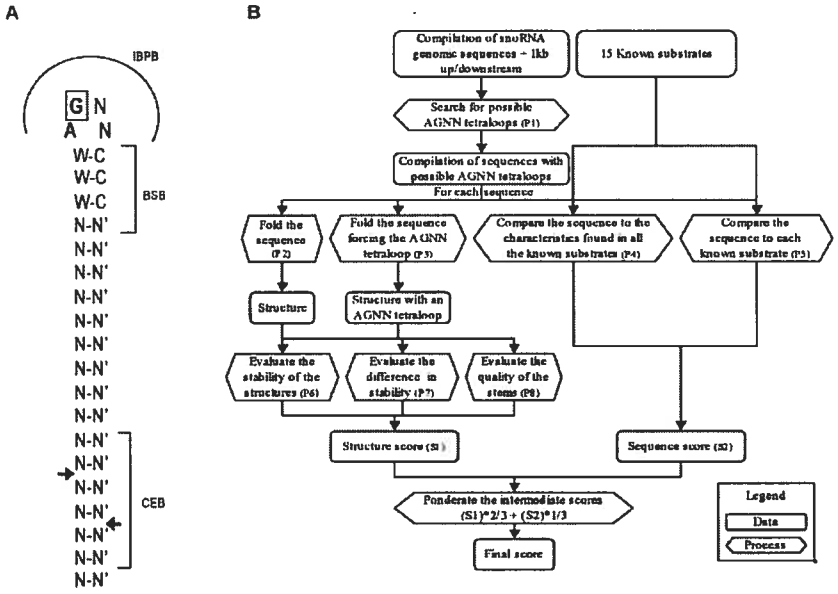


Figure 4.6 – (A) : consensus recherché pour le site de clivage à l'enzyme RNT1P ; (B) : diagramme de recherche. Figure 1 de [18].

leur séquence, et un score s_2, $s_2 \in [0, 1]$ est déduit de cette analyse. Ce score est proche de 1 pour les séquences qui ressemblent aux substrats connus. Le score final tient compte de ces deux scores, et un seuil peut être appliqué afin de filtrer les occurrences.

Les principaux désavantages de la méthode proposée sont que celle-ci ne recherche tout d'abord que des AGNN tetraloops, et que le manque de caractéristiques sur l'hélice rend difficile la gestion de l'ensemble de faux positifs (FP) obtenus. En effet, la recherche du descripteur fourni à RNAMOTIF sur un génome entier retourne un nombre important d'occurrences (74215 tetraloops), et dans une optique de réduction de FP snoRNAs, cette approche n'est pas la meilleure. Le choix du descripteur explique bien pourquoi la base de données n'est pas un génome entier, mais des sous-séquences bien définies (régions près de snoRNAs connus). De plus, l'autre désavantage est l'utilisation de trois façons différentes pour effectuer le repliement de structure secondaire, et trois autres façons d'évaluer la qualité de la structure secondaire obtenue à l'aide du logiciel Vienna RNA Package. Cette méthodologie induit un biais vers les hélices ne possédant pas ou peu de bulges (par minimisation d'énergie libre). On ne parvient donc pas à gérer les motifs

présentant des « long-range interaction » (voir section 2.6).

4.6 Conclusion

Les méthodes généralistes sont utiles lorsqu'aucune méthode spécifique de recherche n'existe. Généralement, ces méthodes nécessitent une bonne connaissance du logiciel et de la molécule à rechercher. Elles sont parfois utilisées dans des approches de recherche spécifiques afin d'accélérer le processus.

Les méthodes spécifiques présentées, bien que ne recherchant pas le même type de molécule, partagent plusieurs caractéristiques communes : caractérisation d'éléments conservés, recherche de structures secondaires. De plus, trois de celles-ci [15, 17, 18] utilisent des logiciels de repliement de la structure secondaire, dont on peut critiquer la pertinence biologique (voir section 4.2.2) et qui augmentent grandement le temps de calcul. La complexité de ces méthodes ($O(n^3)$) rend le logiciel lourd. Nous voulons nous démarquer de l'approche snoSCAN par l'utilisation de nouvelles contraintes sur le snoRNA.

Pour ce qui est de la recherche spécifique de tetraloop, dans notre optique de recherche qui est de relaxer les contraintes sur les snoRNAs mais d'y inclure le site de clivage à l'enzyme RNT1P, la méthode proposée dans [18] n'est pas appropriée car elle n'introduit pas assez de contraintes sur le motif tetraloop. Cette méthode a donc été écartée du projet pour la recherche de snoRNAs mais certaines portions du projet s'en inspirent. Les prédictions faites par mon approche sont mises en relation avec les résultats de cette méthode à la section 6.

Les méthodes présentées sont utiles dans le cadre de ce mémoire, parce que, dans certains cas, on s'inspire de celles-ci (méthodes présentées en section 4.3.1, 4.5.1 et 4.3.1), et que certaines nous permettent de comparer nos résultats au chapitre 6 : les méthodes présentées au sections 4.5.1 et 4.5.3.

CHAPITRE 5

NOUVELLE MÉTHODE DE RECHERCHE DE SNORNAS DE TYPE C/D

5.1 Introduction

L'objectif de ce travail est de découvrir de nouveaux gènes de snoRNA dans le génome de la levure *Saccharomyces cerevisiae*, non répertoriés dans les bases de données. Étant donné que le nombre de sites de méthylation sur l'ARN ribosomal est de 55 et que 51 de ceux-ci sont attribués à 41 snoRNAs, le nombre maximal de snoRNAs restant à découvrir serait de 4. Mais, comme il a été souligné en section 2.5, on peut s'attendre à en obtenir plus puisque 46 sont répertoriés ; cela signifie donc que les snoRNAs de type C/D ne sont pas circonscrits qu'au seul rôle de méthylation de l'ARNr.

La méthode développée a uniquement pour but de découvrir un certain nombre de snoRNAs manquants, en utilisant des critères de recherche différents des méthodes précédentes. Comme il a été mentionné dans la section 4.5.1, la méthode existante s'appuie principalement sur la recherche du motif ASE. Il est donc obligatoire de connaître l'ARN à méthyler. Cette méthode ne peut donc pas retrouver des snoRNAs dont on ne connaît pas la cible. Ma méthode quant à elle n'est pas destinée à une utilisation automatisée permettant de retrouver tous les snoRNAs dans un génome donné, mais ne possède pas ce handicap. Les gènes les plus prometteurs aux yeux de ma méthode (ayant la plus forte probabilité de correspondre à des snoRNAs) pourront ensuite être testés en laboratoire.

La particularité de notre recherche par rapport aux méthodes antérieures consiste à combiner la recherche des boîtes C/D caractérisant le transcrit du snoRNA avec la recherche du site de reconnaissance de l'enzyme RNT1P, devant se trouver en amont ou en aval du snoRNA. Ce site de reconnaissance est caractérisé par un signal tige-boucle qu'on désignera par « tetraloop ».

Comme il a été précisé à la section 4.5, la description la plus précise pour le motif tetraloop doit être disponible afin de trouver les occurrences véritables dans le génome. La majeure partie de la procédure développée a consisté à étudier ce motif et trouver des

façons de réduire le nombre de candidats tetraloops afin de réduire par le fait même le nombre de candidats snoRNAs.

L'étude du signal de reconnaissance à l'enzyme RNT1P permet d'élargir les horizons de recherches. Bien que la méthode développée n'ait été testée que pour retrouver des snoRNAs de type C/D, d'autres mécanismes cellulaires sont régis par cette enzyme et l'acquisition de connaissances pour ces mécanismes pourrait profiter de la présente expertise. Par exemple, il a été démontré par l'équipe du Dr. Abou Elela [8], que l'enzyme RNT1P serait impliqué dans la dégradation de l'ARNm d'un répresseur du cycle du glucose chez la levure, Mig2p. Cette enzyme serait aussi impliquée dans la dégradation d'un ARNm associé à la télomérase.

Je commencerai par décrire toutes les étapes de notre approche, puis je justifierai l'utilisation des différents paramètres de recherche.

5.2 Approche générale

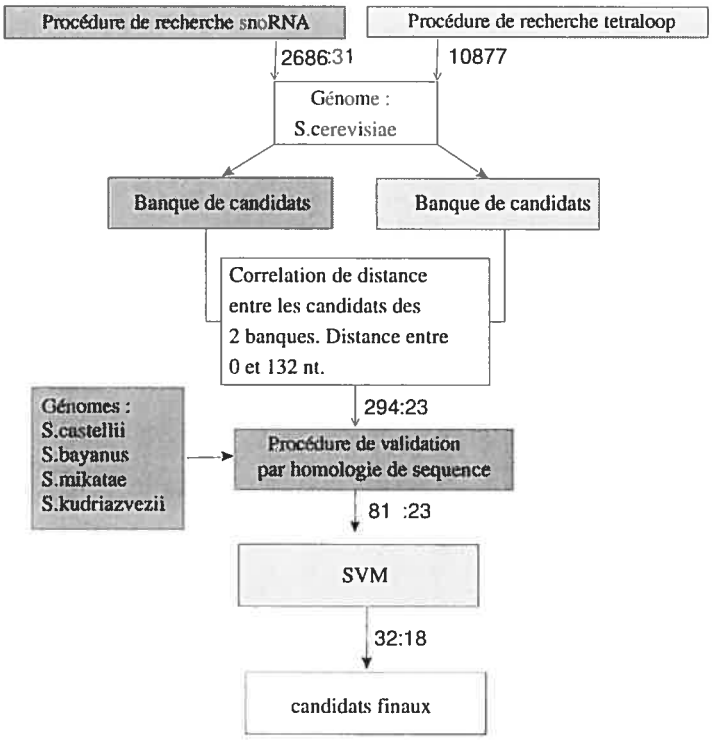


Figure 5.1 – Procédure globale de recherche élaborée pour trouver de nouveaux snoRNAs. Les nombres sur les flèches indiquent le nombre de candidats snoRNAs restants après l'étape de filtration, ceux écrits à la droite du " :" sont les snoRNAs TP.

La procédure de recherche est explicitée à la figure 5.1. Dans un premier temps, la séquence génomique (*Saccharomyces cerevisiae*) est parcourue deux fois, pour la recherche de tous les sites potentiels de snoRNAs et tous les sites potentiels de sites RNT1P. Les deux listes de positions sont ensuite comparées et filtrées pour ne garder que les paires de positions (une correspondant à un site snoRNA et une à un site RNT1P) se trouvant à une distance raisonnable (voir section 2.6). Une procédure de validation par homologie de séquence consiste ensuite à comparer les séquences des candidats snoRNAs de chaque paires avec leurs séquences homologues dans quatre autres espèces de levures. À la fin de cette procédure, on ne garde que les candidats qui ont une similarité suffisante avec au moins 3 des 4 autres homologues. Finalement une méthode d'appren-

tissage machine SVM est utilisée afin d'éliminer les candidats les moins prometteurs. Ce dernier filtrage se base sur le signal RNT1P. Dans la suite, nous détaillons chacune de ces étapes de recherche.

5.2.1 Procédure de recherche des séquences génétique de snoRNA

La recherche de snoRNA de type C/D est présentée à la figure 5.2. Elle consiste en la recherche de la boîte C et de la boîte D séparées d'une distance pouvant varier de 42 à 200nt. La recherche de la boite C (respec. de la boite D) consiste à rechercher le consensus $[A,G]UGAUGA$ (respec. $CUGA$) sans erreurs. Finalement, une hélice terminale de longueur 4 à 8 paires de bases doit être présente en amont des boîtes C et D.

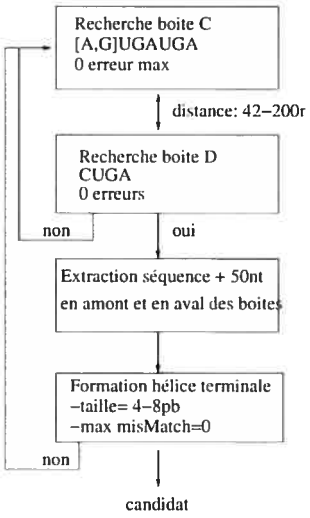


Figure 5.2 – Procédure de recherche des snoRNA de type C/D.

Pour chaque occurrence de la boîte C commençant à la position i, la boîte D est recherchée dans la sous-séquence délimitée par les positions $i+42$ et $i+200$. Les séquences consensus des boîtes C et D sont recherchées à l'aide de l'algorithme de Knuth-Morris-Pratt [9] en temps linéaire. La recherche des deux signaux se fait donc en temps $O(n+m)$ où n est la taille de la séquence génomique parcourue, et m le nombre d'occurrences de la boîte C.

La recherche d'une hélice terminale se fait en utilisant l'algorithme d'alignement lo-

cal Smith-Waterman [50]. On commence d'abord par extraire 50nt en aval de la boîte C (respect. de la boîte D). On inverse la séquence en aval de la boîte C puisque l'on souhaite former un hélice, ce qui requiert un repliement dans l'espace. On doit ensuite redéfinir les notions de « Match » et de substitution. Ainsi, un « Match » est défini selon les appariements WC et le wobble : si à une position i de l'alignement nous avons un A, alors il faudra un U dans l'autre séquence. Une substitution est définie par l'ensemble de tous les autres types d'appariements. Finalement une insertion / suppression d'un nucléotide dans l'une des séquences alignées introduit un « gap » dans l'alignement, donc un « bulge » dans l'hélice ainsi formée (voir section 2.3). Cette redéfinition des opérations de l'algorithme permet de trouver la structure secondaire ayant le moins d'erreurs (la portion 5' est la plus similaire dans sa relation d'appariement avec la portion 3') dans une séquence.

Les relations de récurrence utilisées pour l'algorithme d'alignement local sont présentées ici avec les valeurs de paramètres suivantes : $indel = -9$, $mismatch = -5$, $match = 1$.

$$D(i,0) = 0 \; pour\, tout \; i, 0 \leq i < 50$$

$$D(0,j) = 0 \; pour\, tout \; j, 0 \leq j < 50$$

$$D(i,j) = max[0, D(i-1,j)+indel, D(i,j-1)+indel, D(i-1,j-1)+p(i,j)] \text{ pour tout } i,j > 0$$

où

$$p(i,j) = \begin{cases} match & si\ (i,j) \in \{(A,U),(U,A)(G,C),(C,G),(U,G),(G,U)\}(app.WC\ et\ wobble) \\ mismatch & sinon \end{cases}$$

5.2.2 Procédure de recherche du signal RNT1P

On recherche la boucle caractérisée par le motif $[A,G,U]GNN$ (voir section 2.6), suivie de trois appariements de type Watson-Crick. Ce motif doit ensuite pouvoir être allongé de façon à obtenir une hélice de 8 à 9pb. Trois appariements de type « wobble » sont autorisés dans la portion terminale de l'hélice, c'est-à-dire la portion qui est à plus

de 3pb de la boucle, ainsi que deux erreurs d'appariements.

Le choix de ces caractéristiques est guidé par les différentes études sur l'interaction de RNT1P avec son substrat (voir section 2.6). Le motif a été défini de façon à être très restrictif en ce qui a trait au nombre de candidats retrouvés, et le choix des paramètres sur l'hélice sera discuté dans la section 5.3. Le motif recherché est donné à la figure 5.3.

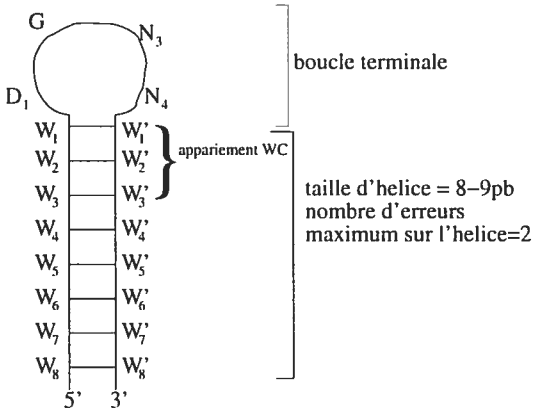


Figure 5.3 – Tige-boucle correspondant au site de reconnaissance de RNT1P (tetraloop). Pour tout indice i, "$W_i - W_i'$" désigne un appariement.

La recherche du motif $DGNN; D \in [A,G,U]$ suivi des trois premiers appariements est effectuée de la façon suivante : on commence par expliciter la liste de tous les mots possibles de taille 10 (4nt pour la boucle et 3 appariements) répondant aux contraintes exigées. Cette liste est ensuite recherchée dans la séquence génomique en utilisant l'algorithme de Aho-Corasick (algorithme de recherche multiple). Cet algorithme permet de localiser l'ensemble des m mots de la liste dans le génome en ordre de complexité $O(m+n)$, où n est la taille du génome. La suite de la recherche consiste en un allongement de l'hélice du motif de façon à obtenir 8-9pb. On fait donc un alignement global, dont la formule de récurrence est présentée ici :

$$D(0,0) = 0$$

$$D(i,0) = D(i-1,0) + \textit{indel pour tout } i, 1 \leq i \leq 8$$

$$D(0,j) = D(0,j-1) + \textit{indel pour tout } j, 1 \leq j \leq 8$$

$$D(i,j) = min[D(i-1,j)+indel, D(i,j-1)+indel, D(i-1,j-1)+p(i,j)] \text{ pour tout } i,j>0$$

où

$$p(i,j) = \begin{cases} match & si\ (i,j) \in \{(A,U),(U,A)(G,C),(C,G),(U,G),(G,U)\}(app.\,WC\ et\ wobble) \\ mismatch & sinon \end{cases}$$

avec la définition des opérations suivante : $indel = 5$, $mismatch = 9$, $match = 1$. Étant donné qu'on veut une hélice de taille au plus 9, et que les trois premiers appariements sont déjà définis, on regarde des séquences de taille 6 avec une possibilité de 2 indels à la fin, ce qui donne 8nt à parcourir. l'algorithme prend donc en entrée les séquences en amont et en aval du motif, celle en aval ayant été inversée afin de trouver la meilleure hélice.

Une dernière étape consiste à éliminer les candidats ayant des bases ou des appariements particuliers. Cette étape a été possible grâce à l'analyse des sites RNT1P rendus disponibles par le laboratoire du Dr. Abou Elela [21]. Nous avons découvert des constantes dans les motifs clivés expérimentalement. Elles sont résumées dans le tableau 5.1. J'ai pu inférer les règles 5 et 6 à l'aide de la statistique d'information mutuelle (voir section suivante). Ces règles contribuent à réduire de 151 le nombre de FP snoRNAs à l'étape 2 (voir figure 5.1), tout en ne générant pas de FN. Les règles 1 à 3 ont été observées par l'équipe du Dr. Abou Elela, la règle 3 ayant été ajoutée au filtre après une inspection manuelle des candidats passant à travers toute la procédure (voir figure 5.1). Il a été montré expérimentalement que les motifs ne respectant pas ces règles ne sont pas clivés *in vitro*. Ces règles, en plus ne ne pas induire de FN, contribuent à la filtration de 350 FP. La règle 4 a aussi été ajoutée après une inspection manuelle des candidats restants ayant passé au travers de la procédure ; elle élimine 100 FP. Ces chiffres viennent montrer l'importance des règles de composition de séquence : on parvient à éliminer un nombre important de candidats tout en respectant notre ensemble de vrais tetraloops.

Inférence des règles à l'aide de la statistique d'information mutuelle

La statistique MI (information mutuelle) [55] a été utilisée afin de trouver les positions co-variantes dans la structure secondaire du signal RNT1P (voir figure 5.3).

Supposons que l'on dispose de n séquences codantes de la même famille d'ARN. Pour pouvoir extraire de l'information de co-variation de ces séquences, il est primordial d'avoir un alignement multiple qui soit fiable. Dans le cas du signal RNT1P, il est facile d'obtenir un bon alignement des échantillons grâce à un ancrage autour de la séquence de la boucle avec ses trois appariements proximaux.

Soit $\mathscr{A}$ un alignement multiple de n séquences codantes. Soient i et j deux positions différentes de l'alignement (deux colonnes), et x et y deux nucléotides quelconques. Nous avons besoin des deux définitions préliminaires suivantes :

1. La fréquence relative F_x du nucléotide x à la colonne i est définie par $F_x = \frac{f_x}{n}$ où f_x est le nombre d'occurrences de x à la colonne i.
2. La fréquence jointe $F_{x,y}$ d'un nucléotide x à la position i et d'un nucléotide y à la position j est définie par $F_{x,y} = \frac{f_{x,y}}{n}$ où $f_{x,y}$ est le nombre d'occurrences de x (respect. y) à la colonne i (respect j).

La co-variation des positions i et j est donnée par la formule suivante :

$$MI_{i,j} = \sum_{x,y} F_{x,y}\, log_2(\frac{F_{x,y}}{F_x \times F_y}),$$

où x et y représentent toutes les paires de nucléotides possibles. Ainsi, pour un couple de colonnes (i, j) présentant une forte co-variation, la formule donnera un score élevé puisque la fréquence jointe des deux nucléotides co-variants sera plus grand que le produit de leurs fréquences disjointes. On aura donc un rapport log important pour les deux nucléotides co-variants, et un rapport log très faible pour les autres types de nucléotides. Dans le cas d'une co-variation minime, on peut s'attendre à avoir autant de tous les types de paires de nucléotides, ainsi le ratio fréquence jointe sur fréquence disjointe tendra vers 1, et donc vers 0 lorsque mis en logarithme.

Une fois les scores obtenus pour chaque paire possible de positions, on doit trouver

une façon de définir les scores significatifs. Tout d'abord, il est nécessaire de modifier le score de MI afin que ce ne soit pas juste les scores correspondant à des appariements qui soient pris en compte. Pour ce faire, le score de MI est multiplié par 1 – P , où P correspond à la fréquence jointe des nucléotides co-variants formant un appariement. On obtient donc des scores très près de zéro pour les nucléotides responsables des appariements canoniques du motif. La figure 5.4 est une représentation centrée-réduite de la population de score. L'abscisse indique les scores discrétisés en tranche de 0,05 et l'ordonnée indique le nombre de paires de positions dont les scores tombent dans l'intervalle donné. On a décidé de conserver, parmi les sept scores les plus élevés, ceux qui ne généraient pas de FN snoRNAs, et on en a tiré 2 règles à l'oeil (règles 5 et 6 du tableau 5.1).

Règles d'élimination
1) $N_1GN_3N_4 \neq GNRA,\ R \in \{A,G\}$
2) $\exists i, 1 \leq i \leq 3 \mid (W_i, W_i') \in \{(G,C),(C,G)\}$
3) Pas d'insertions du côté 3' dans les cinq premiers appariements
4) Au plus une insertion du côté 5' dans les six premiers appariements
5) $N_1GN_3N_4 = DGUU \implies W_2 \neq R$
6) $W_4 = G \implies W_5' \neq R$

Tableau 5.1 – Règles inférées sur le motif RNT1P. Les candidats ne respectant pas celles-ci sont éliminés. W indique une base dans l'hélice, N une base dans la boucle, l'indice se référant à la figure 5.3.

5.2.3 Corrélation de distance

Les deux banques de candidats générées par la recherche du signal snoRNA et du signal RNT1P sont mises en relation de distance sur le génome : seuls les snoRNAs possédant un signal RNT1P à moins de 132 nucléotides en amont ou en aval sont retenus pour la suite de la procédure. Selon le Dr. Abou Elela, cette distance devrait être d'environ 100nt à partir du point de clivage de la tetraloop, mais, comme il sera vu en section 5.3, cette distance est la plus spéficique et la plus sensible à notre ensemble de séquences.

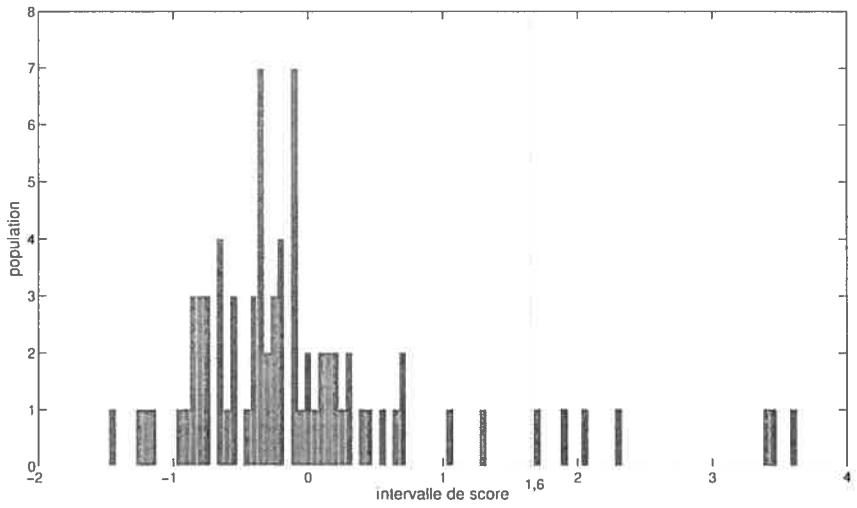


Figure 5.4 – Distribution des scores centrés-réduits pour chaque paire de nucléotides résultant de la statistique d'information mutuelle. Les scores supérieurs à 1,6 sont à la droite de la ligne pointillée.

5.2.4 Procédure de validation par homologie de séquence

Afin d'éliminer un certain nombre d'artefacts de nos banques de candidats, on recherche une homologie de séquence de chacun de nos candidats snoRNAs chez les quatre espèces voisines de *Saccharomyces cerevisiae* mentionnées à la figure 5.1 et dans la section 2.8. Cette portion de la méthode est effectuée en utilisant l'outil BLAST (voir section 3.3.4) [11]. Les seuils de filtration choisis sont : score de probabilité E-value inférieur à 6×10^{-17}, et homologie de séquence présente chez au moins trois des quatre espèces voisines. Ces paramètres ont été choisis car il a été déterminé qu'à ces seuils, aucun FN (faux négatifs snoRNAs) n'est généré alors qu'une bonne filtration de candidats snoRNAs est effectuée. Ainsi, comme il a été mentionné en section 2.5, les snoRNAs ont une taille qui varie entre 70 et 250nt, pour une moyenne de 124nt sur nos 31 TP (vrai positifs snoRNAs) répertoriés (voir recherche de snoRNAs, figure 5.1 et table I.1 de l'annexe). Un score de E-value 6×10^{-17} correspond à une identité de séquence de plus de 80% et dont la fraction de « gap » correspond à moins de 2% de la séquence. Par exemple, une séquence qui possède 94% d'identité, avec une composition équivalente de

tous les nucléotides et dont la longueur est de 54 obtient un score E-Value de 6×10^{-17}. On comprend que si on demande un score plus faible encore, on pénalise les snoRNAs dont la séquence est inférieure à 100nt, car, même en ayant une identité de séquence excellente, ils ne pourront pas avoir un score aussi faible que les candidats à longues séquences, et seront éliminés par la méthode. Le score est donc une mesure indirecte de l'identité de séquence et de la taille de la séquence. Tous les « hits » recensés ayant ce score ont une taille moyenne de 86nt.

Le choix de garder les séquences ayant un homologue dans seulement trois des quatre espèces considérées est partiellement justifié par le fait qu'une des quatre espèces, *Saccharomyces castellii*, peut être considérée comme « outgroup » (voir figure 2.7). En effet, alors que les autres font partie du groupe *Saccharomyces sensu stricto*, et ayant une date de spéciation évaluée à 20 millions d'années avec *Saccharomyces cerevisiae*, *Saccharomyces castellii* est une parente qui serait éloignée de plus de 20 millions d'années (voir figure 2.7). Le but d'utiliser une levure plus éloignée est d'éliminer les candidats snoRNAs qui auraient une bonne identité de séquence dans les quatre levures ; cette information pourrait indiquer que le candidat est localisé dans l'intron d'un gène essentiel dans le cas où le candidat possède une bonne homologie même chez *Saccharomyces castellii*.

5.2.5 SVM

L'algorithme SVM (Support Vector Machine) a été utilisé afin de réduire le nombre de faux positifs de snoRNAs. L'analyse porte sur le signal tetraloop de ces snoRNAs. L'objectif est d'éliminer les candidats qui sont le moins susceptibles de correspondre à de vrais signaux RNT1P. SVM est un algorithme d'apprentissage permettant d'extraire les caractéristiques d'une famille de séquences à partir d'un ensemble d'entraînement. Dans notre cas, l'ensemble d'entraînement contient les séquences RNT1P validées en laboratoire (séquences clivées expérimentalement par l'équipe du Dr. Abou Elela). L'ensemble d'entraînement doit aussi contenir un certain nombre de contre-exemples, c'est-à-dire des séquences que l'on sait ne pouvant pas être clivées par l'enzyme RNT1P. Plus précisément, cet ensemble est constitué d'un ensemble de *vrais positifs* et d'un ensemble

de *vrais négatifs* définis comme suit :

- Ensemble de *vrais positifs* : séquences qui sont reconnues par l'enzyme RNT1P et clivées. Nous avons à notre disposition 45 séquences testées en laboratoire (équipe de Sherif Abou Elela) et reconnues pour être clivées par RNT1P [22].
- Ensemble de *vrais négatifs* : séquences qui ne sont pas clivées car elles n'ont pas les motifs nécessaires à la reconnaissance de l'enzyme. Nous avons choisi de considérer comme vrais négatifs un ensemble de séquences prélevées aléatoirement du génome de *Saccharomyces cerevisiae* qui se replient en hélices dont la boucle est de taille variant de 3 à 6. Le nombre total de ces séquences est de 292.

Une description plus complète de la nature des ensembles est donnée en annexe I.2. Le SVM tente de définir une marge géométrique dans l'espace des caractéristiques des échantillons. L'idée est de pouvoir faire une classification binaire, c'est-à-dire de séparer du mieux possible les vrais positifs des vrais négatifs. « Mieux » signifie que cela ne doit pas se faire au détriment de la capacité de généralisation du modèle. Ainsi, un modèle qui sépare sans erreurs peut apprendre des caractéristiques qui semblent importantes à la classification des échantillons utilisés, mais qui, sur d'autres échantillons, peuvent s'avérer sans intérêt et même conduire à une mauvaise classification.

L'*erreur de généralisation* est une mesure permettant de vérifier la fiabilité de la méthode lorsqu'elle est utilisée sur de nouveaux échantillons. L'*erreur d'apprentissage* reflète le degré d'apprentissage effectué sur les échantillons d'entraînement. Généralement, une faible erreur d'apprentissage est synonyme de « overfitting », c'est-à-dire de sur-apprentissage sur les données d'entraînement, entraînant une mauvaise généralisation sur de nouvelles données.

Une méthode de validation croisée simple [46] a été utilisée afin de paramétrer le modèle d'apprentissage de façon optimale. Cette approche est utilisée dans le cas où les échantillons d'entraînement et de test sont peu nombreux. Le but de celle-ci est de vérifier si les hyper-paramètres choisis du modèle d'entraînement sont suffisamment bien ajustés pour gérer et calquer la distribution désirée. Cette méthode consiste à segmenter l'ensemble de recherche en k blocs, d'utiliser $k-1$ blocs pour l'entraînement du modèle, et le bloc restant pour tester le modèle. Les résultats d'efficacité du modèle deviennent

donc des moyennes de k résultats.

Dans le cas du SVM, une double boucle itérative sur cette procédure a été appliquée afin de trouver la meilleure combinaison de C et de σ (voir section 5.3). Dans le cas de l'implémentation du SVM utilisée, libSVM [6], l'option « -v » définissait la validation croisée simple, le nombre de blocs k a été fixé à 10. Pour une description plus complète du modèle SVM et de ses paramètres, voir l'annexe I.3.

Étant donné que le SVM ne peut traiter l'information biologique (séquence, structure de la molécule), une étape de transformation du signal vers des valeurs numériques est requise. Cette étape est décrite ci-dessous.

Transformation du signal biologique

La méthode de transformation du signal biologique est basée sur les articles de Asai [49] et de Xue [27]. Elle consiste à calculer des fréquences relatives de certains motifs sur la structure tetraloop (voir section 5.2.2).

Les ensembles d'échantillons décrits à la section 5.2.5 sont codés de la façon suivante : pour chaque échantillon et chacun de ses nucléotides, on code la base et les n éléments suivants dans la structure secondaire : 1 si apparié, 0 sinon. Étant donné qu'il y a 4 nucléotides possibles, et que le codage de la structure secondaire est binaire, pour une taille du motif $n = 3$, un motif étant ici défini comme un n-tuplet combinant l'information de séquence et de structure, le nombre de valeurs possibles est $4 \times 2^3 = 32$. Après avoir traversé la séquence et rempli le vecteur des motifs selon la façon décrite en figure 5.5, on normalise chacune des cases par la taille de la séquence. Un échantillon est donc représenté, pour $n = 3$ par un vecteur de motifs à 32 cases avec des valeurs dans l'intervalle $[0, 1]$. L'aspect important de la transformation de l'information concerne le décodage de la structure secondaire. La structure secondaire est codée en bits, la valeur correspondante doit donc être convertie en valeur décimale pour être additionnée avec le décalage associé à la lettre. La figure 5.7 montre que la valeur $n = 4$ permet d'atteindre le meilleur ratio sensibilité/spécificité et c'est pour cela qu'elle a été préférée.

Après le codage, il est nécessaire de créer une fonction de correspondance dans l'espace de Hilbert, nous utilisons la représentation plus simple de noyaux. La fonction

de noyau permet de définir la distance entre 2 échantillons et est requise par l'algorithme d'apprentissage pour définir une marge géométrique entre les différentes classes. Le noyau utilisé est « Radial Basis Function » (RBF). Celui-ci a été préféré aux autres fonctions de noyau (linéaire, polynomial et sigmoïde), parce qu'il offrait de meilleurs résultats du ratio sensibilité/spécificité.

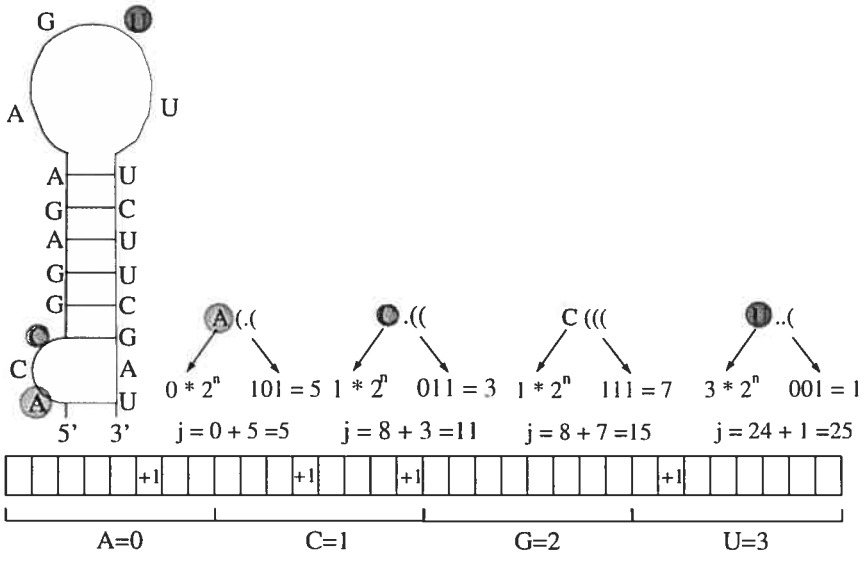


Figure 5.5 – Méthode de transformation du signal biologique en signal numérique. *n*= le nombre de bases successives à considérer. Codage des symboles : A=0, C=1, G=2, U=3, (= 1, . = 0 .

Voici le pseudocode utilisé pour calculer la fréquence des possibilités de mots sur une séquence d'ADN S, avec un tableau représentant la structure secondaire T dans l'alphabet [(,), .] et une taille de motifs n.

(**procédure**) $findSpectrum(S, T, n)$

1: $tableDeTuplets = new\ Table[1...4 \times 2^n]$ {Une table contenant les fréquences des mots}
2: **for** i=0 to S.taille **do**
3: $offset = 0$
4: **if** S[i]=A **then**
5: $offset = 0$

6: **else if** S[i]=C **then**
7: $offset = 1$
8: **else if** S[i]=G **then**
9: $offset = 2$
10: **else if** S[i]=U **then**
11: $offset = 3$
12: **end if**
13: $offset = offset \times 2^n$
14: $pattern = findPairingSchema(T, i, n)$
15: $tableDeTuplets[pattern + offset] += 1$
16: **end for**

La fonction appelée pour calculer le codage d'un motif, "findPairingSchema", prend en entrée la structure secondaire T telle que définie précédemment, une position i dans cette table ainsi que la taille n du mot.

(procédure) $findPairingSchema(T, i, n)$

1: $pattern = 0$
2: **for** j=i to i+n **do**
3: $car = 0$
4: **if** T[i]= (**then**
5: $car = 1$
6: **else if** T[i]=) **then**
7: $car = 1$
8: **else if** T[i]= . **then**
9: $car = 0$
10: **end if**
11: $pattern = pattern + (2^{j-i} \times car)$
12: **end for**

5.3 Choix des paramètres de la recherche

5.3.1 Courbe ROC

La courbe ROC (« Receiver Operating Characteristic »), généralement utilisée en apprentissage machine, est une courbe qui mesure la relation entre la sensibilité et la spécificité d'une méthode (voir section 2.9). Typiquement, un algorithme de classification aléatoire donnerait une courbe pour laquelle ces deux fractions restent proportionnelles peu importe les différentes conditions de recherche imposées à l'algorithme, donnant ainsi une droite de pente $m = 1$. Un algorithme performant devrait rapidement tendre vers une sensibilité excellente (proche de 1), tout en gardant un nombre de faux positifs bas, donc une spécificité proche de 1. La partie intéressante dans un graphe ROC est le coin supérieur gauche, parce que ce coin indique les meilleurs ratios sensibilité/spécificité (voir Figure 5.6). La courbe ROC a été utilisée dans le choix des paramètres de recherche du signal de reconnaissance à l'enzyme RNT1P (section 5.2.2) ainsi que dans le choix des paramètres n, γ et C pour l'analyse par SVM (section 5.2.5).

5.3.2 Choix des paramètres relatifs au motif tetraloop

Pour le choix des paramètres impliqués dans la recherche du tetraloop (section 5.2.2), il a été nécessaire de prendre en compte le ratio sensibilité/spécificité, mais en privilégiant la spécificité par rapport à la sensibilité. Étant donné que notre but principal de recherche est de trouver de nouveaux snoRNAs, nous nous soucions surtout de sélectionner les candidats qui ont le plus de chance de représenter de "bons" motifs, c'est-à-dire de réduire le plus possible le nombre de candidats FP générés par la méthode, quitte à perdre certains TP au passage. Si l'on souhaite garder les 31 TP tout au long de la méthode (voir figure 5.1), le nombre de FP générés demeure trop grand après passage dans la méthode SVM (plus de 200 candidats), rendant l'analyse post-traitement impossible. À l'inverse, en spécifiant des paramètres de recherche assez restrictifs, il est possible de réduire de beaucoup le nombre de FP.

Une étude du ratio sensibilité/spécificité en fonction de tous les paramètres de recherche a été effectuée. Les dix paramètres étudiés sont : nombre maximal de « mis-

match (variant de 0 à (taille de l'hélice - 3)) » et de « wobble (variant de 0 à (taille de l'hélice - 3)) » autorisés pour l'hélice, taille de l'hélice (variant de 3 à 17 paires de bases), distance permise entre le signal RNT1P et le snoRNA correspondant (variant de 50 à 300nt), respect des contraintes statistiques (règles 5 et 6 de la table 5.1), et respect des autres contraintes (règles 1 à 4 de la table 5.1).

Les résultats sont présentés à la figure 5.6. Chaque point de ce graphique représente le résultat obtenu de candidats pour une combinaison donnée des dix paramètres cités plus haut (vecteur de taille 8). Les valeurs de spécificité et sensibilité sont déduites de l'utilisation des formules et en sachant la quantité de candidats de départ (recherche de snoRNA seul dans la figure 5.1). Le graphique représente un ensemble de points obtenus par un échantillonnage aléatoire de cet espace de recherche. Comme il a été mentionné plus haut, le but est d'obtenir le résultat le plus spécifique (le moins de candidats possibles) tout en étant très sensible (très peu de génération de FN). Les meilleurs points sont donc ceux situés dans le coin supérieur gauche (le rectangle indique la zone qui nous satisfait). Le point choisi pour notre recherche est indiqué à l'aide de la flèche dans la figure 5.6. Il correspond aux paramètres suivants : taille 8-9pb pour l'hélice avec au plus deux substitutions et trois wobbles, respect de toutes les contraintes mentionnées, distance maximale inter-moléculaire :132nt.

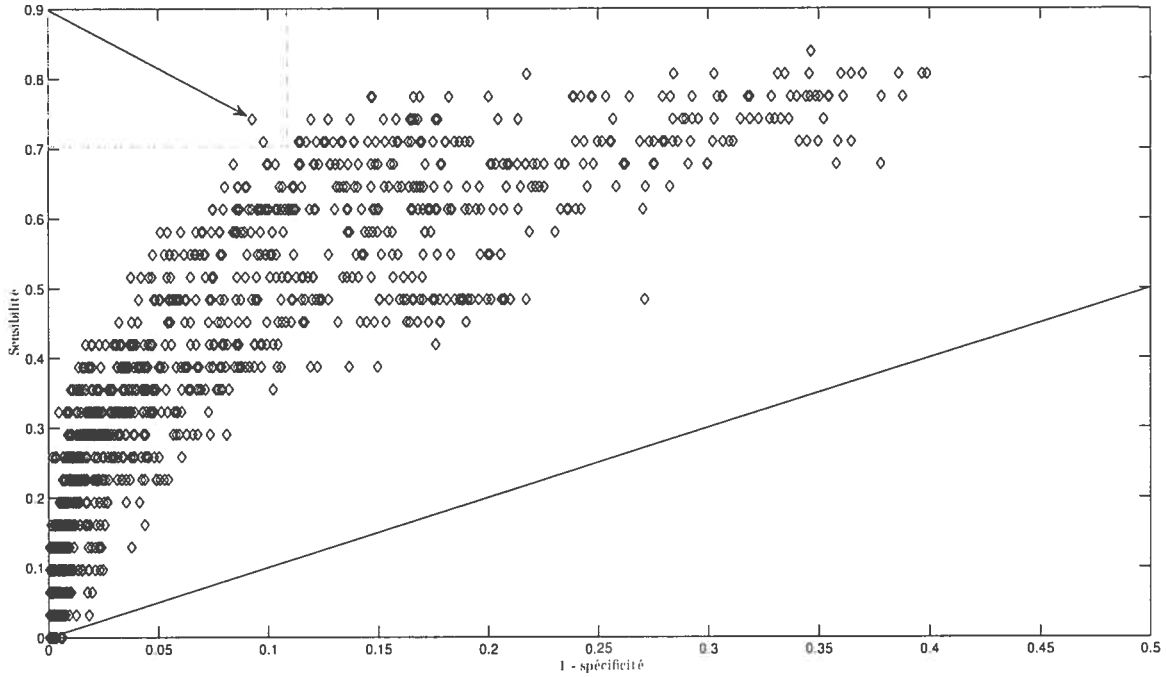


Figure 5.6 – Résultats en spécificité et en sensibilité de la recherche du tetraloop. Chaque point représente une combinaison donnée de tous les paramètres de la recherche. La ligne pleine représente une décision aléatoire, et le rectangle les résultats que l'on juge les meilleurs. Le point sélectionné dans notre recherche est indiqué par la flèche et correspond aux paramètres définis en section 5.2.2.

5.3.3 Choix des seuils de l'étape BLAST

La valeur de E-value utilisée pour la filtration a été déterminée de façon à ne générer que très peu de FN tout en ayant un bon pourcentage de filtration (80%). Au seuil choisi, on ne perd pas de TP, et en le mettant un peu plus faible, la perte en sensibilité ne serait pas compensée par un gain significatif en spécificité. Les différents tests effectués sur ce paramètre tendent à montrer que ce seuil est le plus restrictif qu'on peut choisir sans perdre en sensibilité. Parmi les 31 snoRNAs de départ, 3 seraient éliminés si le seuil était dans la tranche de 6×10^{-17} à 1×10^{-20}. Du fait de leur petite taille, la majorité des snoRNAs ne peuvent obtenir un score inférieur au seuil choisi et c'est la raison qui motive notre choix.

5.3.4 Validation du modèle SVM

Plusieurs valeurs de n, de C et de γ ont été testées pour la transformation du signal (voir section 5.2.5). Une fouille de l'espace de recherche s'est faite avec les intervalles de valeurs suivantes : $0 < C \leq 2000$, $1 \times 10^{-8} \leq \gamma \leq 1$ et $1 \leq n \leq 6$. Le résultat de cette fouille est décrit à la figure 5.7.

La courbe ROC est un argument venant appuyer mes choix de paramètres pour le SVM (voir figure 5.7). De même qu'indiqué en section 5.3.1, les valeurs les plus près du coin supérieur gauche sont données par les paramètres les plus fiables. On remarque que n=4 donne la meilleure valeur lorsque l'on cherche à être au-dessus d'un seuil de sensibilité de 75% (rectangle pointillé). Dans cette optique, n={1,5} semblent être les pires valeurs puisque les résultats de celles-ci oscillent autour de la réponse aléatoire (ligne pleine). Avec les valeurs de paramètres choisies, l'efficacité de validation croisée est à 89,09%. Pour ce qui est du point n=3 étant plus à gauche que le point choisi, mais avec une sensibilité légèrement inférieure, deux arguments motivent notre décision de ne pas le préférer au point choisi. Tout d'abord, les valeurs de C et de γ pour ce point montrent une complexité supérieure : $\gamma = 2 \times 10^{-7}$, $C = 1950$, comparativement à $\gamma = 2 \times 10^{-4}$, $C = 141$ pour $n = 4$ se traduisant par une diminution de 11% en efficacité de validation croisée (89% versus 78%). Ceci s'explique par le fait que le modèle paramétré

à $n = 3$ calque trop les données d'entraînement et ne généralise pas aussi bien la classe de molécule. Les résultats présentés dans la section 6 concernant le SVM sont obtenus avec les valeurs de paramètres suivantes : $n = 4$, $\gamma = 2 \times 10^{-4}$, $C = 141$.

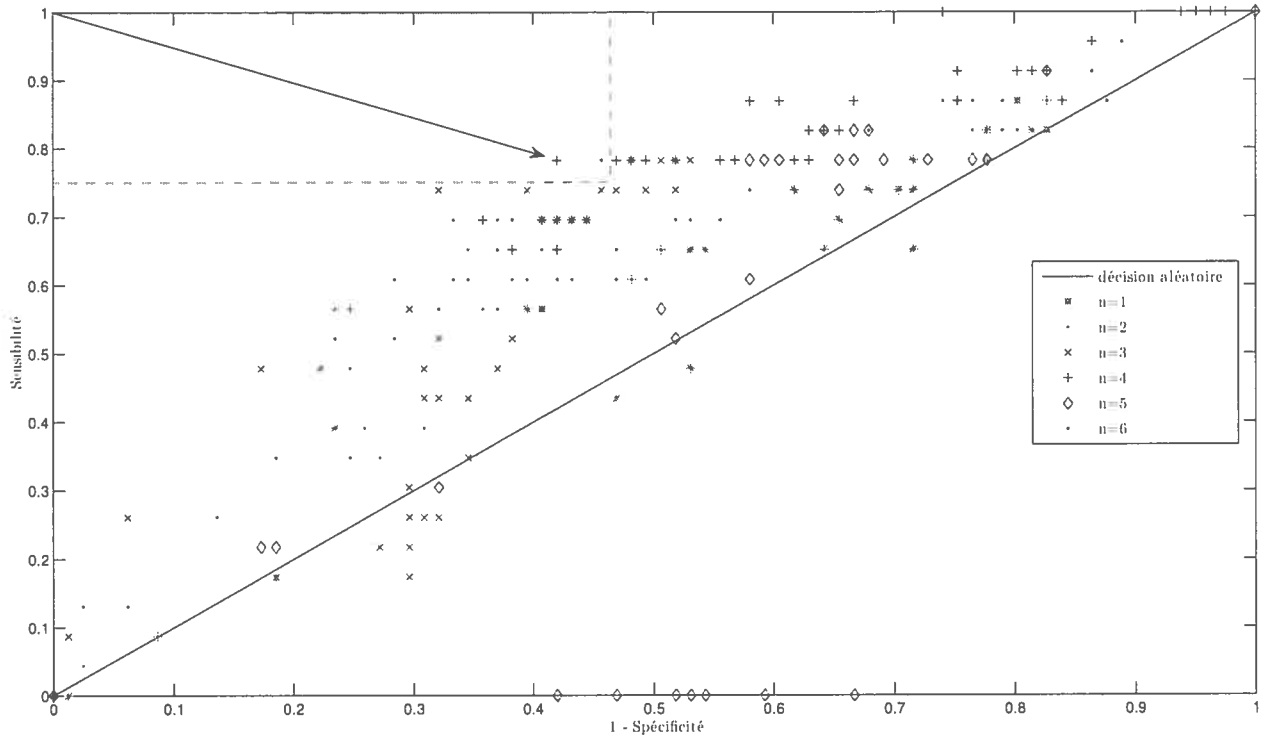


Figure 5.7 – Courbe ROC permettant de vérifier quel ensemble de paramètres est le plus approprié à notre analyse de candidats en regard de l'analyse SVM. Chaque point correspond aux valeurs de sensibilité et de spécificité associé au vecteur de trois paramètres (n, γ, C). La ligne pleine définie la décision aléatoire, le rectangle est un seuil d'acceptation en terme de sensibilité (78% = $\frac{18}{23}$ TP). Le point choisi est indiqué par la flèche et correspond aux valeurs suivantes : $n = 4, C = 141, \gamma = 2 \times 10^{-4}$.

5.4 Conclusion

Comme il sera démontré dans le chapitre suivant, la procédure parvient à cerner un nombre intéressant de candidats, possède un ratio sensibilité/spécificité assez élevé, et nous conforte dans les candidats obtenus. L'étape ultime serait la validation expérimentale des substrats à l'enzyme RNT1P trouvés ainsi que l'inactivation génétique (knock-out) de ces régions afin de voir si le patron de méthylation est modifié dans *Saccharomyces cerevisiae*. Par contre, plusieurs outils sont à notre disposition afin de mieux caractériser notre ensemble final de candidats. La prochaine section souligne les résultats obtenus par ma méthode, discute des faiblesses de celle-ci et explore les outils permettant de discuter de mes candidats finaux.

CHAPITRE 6

RÉSULTATS

6.1 Introduction

Nous présentons dans ce chapitre les prédictions de snoRNAs de type boîte C/D que nous obtenons sur le génome de la levure *Saccharomyces cerevisiae*. Les snoRNAs potentiels seront comparés aux "vrais" snoRNAs, c'est-à-dire les snoRNAs répertoriés dans les bases de données, afin de mieux caractériser mon ensemble final de candidats.

Tout d'abord, la trace des faux positifs (FP) et des vrais positifs (TP) à chaque étape de la recherche sera présentée, suivra ensuite les candidats finaux en relation avec les vrais snoRNAs via trois approches. Nous ferons la comparaison des résultats, et attributions de scores par la méthode snoSCAN [51], ensuite viendra la comparaison des résultats, et attributions de scores sur le signal de reconnaissance par le modèle de recherche développé à Sherbrooke (DRSD : « Dynamic RNT1P Substrate Database » [18]), et finalement, comparaison des résultats en relation avec la concentration cellulaire des transcrits snoRNAs afin de voir si les candidats pourraient être clivés par l'enzyme RNT1P.

Il est à noter que les candidats snoRNAs potentiels se retrouvent dans l'ensemble des FP. Étant donné qu'on estime le nombre de snoRNAs restants à 4, on considère que la majorité des candidats sont effectivement des FP. De plus, l'ensemble de candidats snoRNAs présenté (voir figure 5.1) n'inclut pas les vrais SnoRNAs, nous avons donc 2686 candidats et 31 TP, soit 2717 séquences restantes à l'étape 1.

6.2 Résultats intermédiaires

Nous présentons, dans cette section, les résultats obtenus après chacune des étapes de notre méthode de recherche (Figure 5.1). L'objectif est simplement de fournir au lecteur une idée sur le nombre de candidats obtenus et sur la capacité des caractéristiques utilisées à filtrer la base de données.

Le comportement souhaité à chaque étape de la méthode est de baisser de façon

Étape	Nombre candidats snoRNAs	Nombres vrais snoRNAs répertoriés	spéc. TN / (FP+TN))	sens. TP / (TP + FN)
Recherche snoRNAs	2686	31	/	0.72
Corrélation signal tetraloop	294	23	0.88	0.74
BLAST	81	23	0.72	1
SVM	32	18	0.60	0.78

Tableau 6.1 – Progression du nombre de FP et TP snoRNAs dans notre méthode. La première colonne indique les principales étapes (voir figure 5.1). Les deux colonnes suivantes sont relatives au nombre de FP et TP snoRNAs respectivement. Vient ensuite la mesure de spécificité puis de sensibilité.

significative le nombre de candidats potentiels, sans trop augmenter le nombre de faux-négatifs. Pour ce faire, les résultats en spécificité et sensibilité (tableau 6.1) à une étape i sont calculés en fonction des candidats à l'étape $i-1$ et i. Plus précisément :

- L'ensemble des *vrais positifs* (TP pour "true" positive) est l'ensemble des vrais snoRNAs restants à l'étape i. Au départ, nous considérons comme vrais snoRNAs l'ensemble des 43 snoRNAs répertoriés dans les bases de données et présentés en Annexe (tableau I.1).
- L'ensemble des *faux positifs* (FP) à l'étape i est l'ensemble des snoRNAs trouvés à l'étape i qui n'appartiennent pas à l'ensemble des vrais positifs.
- L'ensemble des *faux négatifs* (FN) à l'étape i est l'ensemble des vrais positifs à l'étape $i-1$ qui ne sont pas reconnus par la méthode à l'étape i.
- L'ensemble des *vrais négatifs* (TN) à l'étape i est l'ensemble des faux positifs à l'étape $i-1$ qui sont éliminés à l'étape i.

La première étape de recherche de snoRNA retourne 2686 FP. Malgré ce nombre élevé, 12 FN sont obtenus, indiquant que les paramètres de recherche du snoRNA seuls sont assez restrictifs. Les faux négatifs générés correspondent généralement à des snoRNAs ayant une boîte C ou D dégénérée. La recherche du signal RNT1P donne 10877 occurrences.

La deuxième étape de la recherche qui consiste à faire correspondre les deux banques de candidats snoRNA et RNT1P. Ceci permet de filtrer de façon significative le nombre de candidats snoRNAs : seulement 294 des 2686 candidats snoRNAs possèdent un signal

RNT1P en amont ou en aval de leur séquence. Cependant, cette étape augmente aussi le nombre de faux-négatifs car le nombre de "vrais" snoRNAs restants passe de 31 à 23.

La procédure de validation par homologie de séquence permet de supprimer des candidats qui n'ont pas de correspondance dans au moins 3 des 4 autres espèces voisines de levure. Cette étape ne supprime aucun vrai snoRNA supplémentaire. Cette étape est presque aussi importante que la précédente puisque le pourcentage de filtration des candidats est le deuxième en importance.

Finalement, l'étape SVM permet de diminuer de moitié les candidats snoRNAs, sans augmenter de façon significative le nombre de faux-négatifs. Les 32 candidats obtenus ayant passé à travers tous les filtres de la méthode sont les plus susceptibles de correspondre à des snoRNAs non encore identifiés. Ces résultats seront présentés de façon détaillée dans les sections qui suivent.

Dans les sections suivantes, les résultats présentés concernent les 32 FP qui restent après l'étape de filtrage avec SVM, ainsi que les 31 TP snoRNAs initiaux (voir figure 5.1 et tableau 6.1). On prend les 31 TP snoRNAs au lieu des 18 passant au travers la procédure pour mieux souligner les caractéristiques des snoRNAs.

6.3 Existence de sites de méthylation pour les candidats via snoSCAN

La fonction la plus documentée des snoRNAs est la 2'-O-méthylation de nucléotides sur l'ARN ribosomal. Le programme snoSCAN [51] introduit à la section 4.5.1 est dédié à la recherche de tels snoRNAs, et les 43 vrais positifs considérés au début de notre procédure de recherche correspondent à ceux retrouvés par cette méthode. Ces snoRNAs sont responsables de la modification de 51 nucléotides sur l'ARN ribosomal. Étant donné que 55 sites existent, il resterait au plus 4 snoRNAs de ce type à découvrir. Cette section explore la possibilité pour notre ensemble de candidats de contenir les snoRNAs manquants. Cependant, étant donné que notre méthode ne considère pas cette contrainte, et que notre objectif est de découvrir des snoRNAs présentant d'autres fonctions non encore étudiées ou des fonctions de méthylation sur des séquences autres que l'ARN ribosomal, le fait de ne pas retrouver de site de méthylation pour un candidat

snoRNA n'est pas un résultat négatif en soit.

Afin de tester la présence de sites de méthylation, nous avons fourni les candidats obtenus par notre méthode comme entrée au programme snoSCAN. Le programme évalue un candidat en calculant un score (fortement influencé par le site de méthylation). Les candidats ayant un score au dessus d'un certain seuil sont acceptés et les autres sont rejetés. Le seuil considéré par T.Lowe pour les vrais snoRNAs détectés par sa méthode (information obtenue sur le site web [4]) est de 12.8, mais tous les autres snoRNAs qu'il retrouve ont un score supérieur à 18. En testant snoSCAN sur nos 32 candidats, seulement 6 ont un score qui dépasse 12.8. Cependant, tous sont de valeur inférieure à 18 (voir table 6.2). De plus, il est à noter qu'aucun des sites de méthylation liés aux candidats restants ne correspond aux sites de méthylation existants. Il est donc peu probable que ce soient des snoRNAs qui méthylent l'ARN ribosomal.

Nom	Score	Site méthylation
psnr1705	14,06	RDN18-1-Am299
psnr346	14,24	RDN18-1-Um1289
psnr1966	14,45	RDN25-1-Cm1693
psnr1150	15,29	RDN25-1-Gm714
psnr1403	15,86	RDN18-1-Um1627
psnr839	15,98	RDN18-1-Am615

Tableau 6.2 – Présentation des 6 candidats (parmi 32) retrouvés par snoSCAN (score supérieur à 12.8). La deuxième colonne donne le score retourné par snoSCAN. La troisième indique le site de méthylation, sous forme simplifiée. Par exemple, le site de méthylation désigné par RDN18-1-Um1289 indique que c'est l'uracile à la position 1289 dans la séquence de la sous-unité ribosomale 18s (petite sous-unité) qui devrait être méthylée. Lorsque le 18 est remplacé par 25, on parle de la grande sous-unité ribosomale.

6.3.1 Évaluation des candidats en fonction du signal RNT1P

Le signal tetraloop de reconnaissance à l'enzyme RNT1P a été étudié par l'équipe de S. Abou Elela qui a développé une méthode empirique de recherche (DRSD, voir section 4.5.3). Parmi les occurrences obtenues par leur méthode, 50 ont été validés expérimentalement [18]. Parmi ceux-ci, 23 sont associés à l'un de nos 31 vrais snoRNAs. Il est intéressant de noter que parmi ces 23 sites tetraloop clivés *in vitro* [18], 18 sont exactement ceux retrouvés par notre programme. Par contre, aucun des signaux testés

expérimentalement n'est relié à l'un de nos candidats et ceci en raison de la méthodologie employée.

Nous avons donc utilisé, pour nos 32 candidats, la méthode empirique développée par l'équipe d'Elela [18]. Selon leur étude les candidats intéressants sont ceux ayant un score DRSD inférieur à 0.3. En particulier, parmi les 23 tetraloops cités ci-dessus, 21 respectent cette règle. Nous classons donc nos candidats tetraloop associés aux 32 snoRNAs en fonction de cette contrainte. La table 6.3 montre que 16 de ces 32 tetraloop ont un score acceptable, et correspondent donc potentiellement à de bons substrats pour l'enzyme RNT1P.

score < 0.3		score > 0.3	
Nom	Score	Nom	Score
psnr1403	0.0708502	psnr1507	0.303763
psnr828	0.111408	psnr1826	0.308706
psnr2406	0.126084	psnr1812	0.313104
psnr351	0.156785	psnr2417	0.322874
psnr907	0.197212	psnr1953	0.345863
psnr1265	0.203019	psnr939	0.363521
psnr624	0.20361	psnr521	0.371227
psnr203	0.208885	psnr839	0.371319
psnr2410	0.209641	psnr1705	0.380577
psnr2118	0.216396	psnr292	0.42404
psnr1163	0.219881	psnr1628	0.44963
psnr1150	0.222432	psnr195	0.504696
psnr728	0.223845	psnr73	0.519386
psnr335	0.235175	psnr2177	0.591303
psnr1966	0.262219	psnr38	0.617927
psnr767	0.283702	psnr346	0.357078

Tableau 6.3 – Ensemble de candidats séparés en fonction du score obtenu par la méthode DRSD (voir section 4.5.3). La section de gauche présente les 16 candidats ayant un score inférieur au seuil, l'autre section présente les 16 dont le score excède le seuil. La première colonne de chaque section indique les noms des candidats, la seconde est relative au programme de recherche de tetraloop développé à Sherbrooke et donne le score de probabilité associé au site trouvé par ma méthode.

Si on met en relation nos candidats ayant un score inférieur à 0.3 avec les candidats de la section 6.3, 3 candidats ressortent (psnr1150, psnr1403 et psnr1966).

6.3.2 Concentration cellulaire des transcrits des candidats finaux

Comme il a été défini en section 2.7, nous souhaitons avoir une différence de concentration de transcrits entre les motifs snoRNA et tetraloop, la concentration du transcrit correspondant au tetraloop étant plus faible que celle du transcrit snoRNA, indiquant que le transcrit snoRNA est clivé puis la portion correspondant au tetraloop est digérée.

La table 6.4 montre le profil de concentration cellulaire de transcrit de mon ensemble final de candidats et des 23 vrais snoRNAs pour lesquels il existe une tetraloop clivée expérimentalement (voir section 6.3.1). Comme on peut le constater dans la table, la majorité des vrais snoRNAs ont une différence de concentration cellulaire des transcrits snoRNAs et tetraloop (20/23). Sept de mes 32 candidats finaux suivent cette tendance.

Si on met en relation les candidats finaux de cette méthode présentant une différence de concentration des transcrits snoRNA et tetraloop avec ceux de la section snoSCAN (section 6.3), un seul candidat ressort : psnr346. Par contre, une mise en relation de ces mêmes candidats finaux avec ceux de la section 6.3.1 permet de voir que 4 candidats ont aussi un score DRSD inférieur à 0.3 : psnr624, psnr767, psnr907 et psnr2118. Nous n'avons donc aucun candidat qui serait considéré comme une bonne prédiction par les trois approches. Néanmoins, nous considérons que les bons candidats retenus des sections 6.3.1 et de cette section sont meilleurs parce que, comme il avait été mentionné lors de la discussion portant sur les candidats par snoSCAN, les sites de méthylation trouvés n'étaient pas de toute façon des sites répertoriés. Parmi les 4 candidats étant retenus par la section précédente et celle-ci, un seul semble aussi avoir une taille de transcrit ayant la même taille qu'un snoRNA mature, il s'agit de psnr624.

Lorsque le transcrit sur lequel se retrouve un candidat final est trop gros et englobe aussi le site tetraloop, alors on n'est pas en présence d'une tetraloop qui agit comme signal de clivage, faisant en sorte que le candidat ne sera pas traité par l'enzyme.

Dans la section suivante, nous analysons les 4 candidats retenus (psnr624, psnr767, psnr907 et psnr2118), en fonction de leur localisation génomique.

Tetraloop clivé expérimentalement					Candidats finaux				
	snoRNA		tetraloop			snoRNA		tetraloop	
Nom	conc.	transcrit	conc.	transcrit	Nom	conc.	transcrit	conc.	transcrit
Snr65	0.00	0	0.00	0	psnr2417	0.00	0	0.00	0
Snr48	5.10	105	0.00	0	psnr767	0.00	0	0.00	0
Snr59	3.65	57	0.00	0	psnr346	0.00	0	0.00	0
Snr58	3.37	81	0.00	0	psnr2118	1.60	1161	0.00	0
Snr69	5.26	105	0.09	521	psnr521	3.17	1289	0.00	0
Snr71	3.75	81	0.20	489	psnr907	0.67	2905	0.00	0
Snr39b	5.97	89	0.28	1793	psnr624	2.14	657	0.36	241
Snr64	5.82	105	0.51	177					
Snr61	5.96	81	0.71	161					
SnRNAZ2	4.62	49	0.79	169					
SnRNAZ3	4.37	113	0.79	169					
Snr39	4.57	81	1.05	233					
Snr70	5.19	161	1.09	113					
Snr51	4.54	81	1.09	113					
Snr66	5.93	89	1.19	233	psnr2410	0.28	2393	0.28	2393
Snr68	5.51	121	1.42	281	psnr1150	0.50	1809	0.50	1809
Snr50	5.83	89	1.65	113	psnr203	0.61	4009	0.61	4009
Snr52	5.49	89	1.71	177	psnr351	0.81	1521	0.81	1521
SnRNAZ7	5.97	81	1.79	409	psnr2406	1.33	1465	1.33	1465
Snr47	5.64	89	1.88	129	psnr195	1.34	2353	1.34	2353
					psnr1265	1.37	3377	1.37	3377
					psnr2177	1.47	1001	1.47	1001
					psnr292	1.79	681	1.79	681
					psnr1705	1.99	2305	1.99	2305
					psnr828	2.01	1241	2.01	1241
					psnr38	2.21	1913	2.21	1913
					psnr1812	0.073	4689	2.51	1617
Snr62	1.61	193	1.61	193	psnr1953	2.52	1521	2.52	1521
Snr63	1.98	161	4.63	233	psnr1163	2.53	2857	2.53	2857
Snr19	5.76	561	5.76	561	psnr1403	2.54	977	2.54	977
					psnr1507	2.66	1625	2.66	1625
					psnr1826	3.05	1521	3.05	1521
					psnr728	3.25	2217	3.25	2217
					psnr1628	3.26	2913	3.26	2913
					psnr839	3.33	2921	3.33	2921
					psnr335	3.58	6473	3.58	6473
					psnr939	3.75	8121	3.75	8121
					psnr73	3.04	273	4.50	305
					psnr1966	5.37	1929	5.37	1929

Tableau 6.4 – Table présentant les 23 vrais snoRNAs dont la tetraloop est clivée expérimentalement (section de gauche) et les 32 candidats finaux (section de droite) en relation avec la concentration cellulaire de transcrit. Les molécules du haut (Snr65 à Snr47 dans la section de gauche, psnr2417 à psnr624 dans l'autre section) montrent une valeur de concentration plus élevée pour le transcrit snoRNA que celui portant le tetraloop, les autres non. La première colonne de chaque section indique le nom du candidat, suivent les deux colonnes relatives au snoRNA, soit la quantité de transcrits et la taille de celui-ci. Les deux dernières colonnes de chaque section concernent le transcrit codant le tetraloop, soit la concentration (conc.) et la taille de celui-ci (transcrit). La résolution pour la taille du transcrit est de 8nt [14].

6.4 Candidats finaux et annotation des régions sur le génome

Parmi les 31 vrais snoRNAs, 3 se situent dans des ORF et 28 sont des gènes non-codants. Étant donné que ces régions ont été annotées ainsi à cause de la présence d'ARN fonctionnels (ARNr, ARNt, snoRNA), il est très improbable de retrouver un candidat dans ces régions.

En ce qui concerne nos 4 candidats décrits précédemment, 3 sont dans des ORF (psnr624, psnr907 et psnr2118), et 1 dans une région non annotée (psnr767). On considère généralement qu'un snoRNA ne peut pas se retrouver dans un ORF parce que ces régions codent pour des protéines. Cependant, le seul snoRNA parmi les 23 dont le signal est clivé expérimentalement et se retrouvant dans un ORF, Snr59, montre une différence de concentration de transcrit (voir table 6.4). Psnr767 peut probablement être écarté puisque sa concentration est nulle (voir table 6.4) et que la région dans laquelle il se retrouve n'est pas annotée. Par contre, on ne peut pas conclure que Psnr767 n'est jamais transcrit parce que les données provenant des « micro-array » dépendent du contexte de l'expérience (type de tissus, type de cellule, etc).

6.5 Discussion

6.5.1 Analyse globale de la méthode présentée

Comme il a été mentionné dans la section précédente, l'approche présentée (voir figure 5.1) permet d'éliminer un grand nombre de candidats (on passe de 2686 FP à 32 candidats potentiels), tout en gardant plus de 60% des vrais snoRNAs. Cette réussite est due au fait qu'on essaie d'être le plus strict à chaque étape de la méthode : la portion concernant la tetraloop a été soumise à une étude exhaustive de son espace de possibilités, l'étape de validation par homologie de séquence a aussi été soumise au même traitement afin de déterminer pour quels seuils de E-value les TP snoRNAs sont perdus.

Par contre, comme les résultats en sensibilité l'indiquent, l'approche présentée n'est pas parfaite ; on perd un nombre important de TP snoRNAs. De plus, la procédure de recherche de snoRNAs est déjà restrictive en elle-même : des 46 snoRNAs connus, seule-

ment 31 traversent ce filtre. Étant donné ce constat, il serait hasardeux de dire que la méthode présentée permet la détection automatique de tous les snoRNAs.

Notre approche à filtres imbriqués n'est pas sans faiblesses, mais, en observant l'ordre des étapes, et surtout la grande différence en sensibilité de chacun des filtres, il est possible de l'améliorer. Une étape est très stricte et ne génère aucun FN snoRNA, il s'agit de la procédure de validation par homologie de séquences. Cette étape devrait donc être mise directement après la recherche des snoRNAs. Il faudrait ensuite vérifier laquelle des autres étapes est la plus stricte et l'ajouter. Finalement, en raisonnant ainsi pour tous les filtres, on obtiendrait une procédure plus fiable parce que moins de TP seraient éliminés. De plus, cette approche relaxerait les critères sur les autres portions du modèle, puisqu'à l'entrée de chaque filtre, nous pourrions avoir un ensemble de candidats qui ressemblerait plus à notre vrai ensemble de snoRNAs.

Le SVM, de par la façon qu'il est entraîné, ne peut être plus sensible peu importe l'endroit où on le met dans la procédure. Le SVM est paramétré par un ensemble d'entraînement qui est étranger à notre ensemble de TP et de FP snoRNAs. En conséquence, Pour obtenir un certain niveau de filtration des FP (ici 60%), on doit nécessairement éliminer des TP. Cette étape est la moins susceptible d'améliorer le modèle et devrait donc être mise en dernier.

L'étape d'élimination des FP par une analyse de leur niveau de concentration de transcrit (voir section 6.3.2) ne dépend pas de l'ordre dans lequel on place les filtres : le nombre de FN généré sera toujours au plus de 4 (Snr19, Snr54, Snr62 et Snr65). Cette étape, qui est donc assez stricte, pourrait être placée après l'étape BLAST, faisant en sorte que le nombre de TP snoRNAs serait encore à $\frac{27}{31}$ après ces deux étapes. Pour des raisons techniques, il n'a pas été possible d'évaluer le nombre de FP qu'on obtiendrait selon ce scénario.

L'étape la plus sensible à l'ensemble de snoRNAs donné en entrée est celle de recherche de tetraloop. Comme on l'a vu dans la section 5.3.2, l'optimisation est faite pour avoir le meilleur gain en spécificité tout en gardant la sensibilité en haut de 78%. Cette étape devrait donc être placée juste avant l'étape SVM puisqu'avec le modèle actuel, nous savons qu'un candidat éliminé par l'étape de recherche tetraloop ne l'aurait pas été

à l'étape de validation par homologie de séquences (voir section 5.3.3). De plus, étant donné que les paramètres choisis à l'étape de recherche tetraloop l'ont été dans le but de réduire le plus possible l'ensemble de FP donné en entrée, en donnant un ensemble plus restreint de FP (si l'étape BLAST est placée entre l'étape de recherche de snoRNAs et celle des tetraloop, on passe de 2686 à 680 FP), on pourrait obtenir un ensemble de paramètres plus souple, et un gain en sensibilité.

En conclusion, la manipulation de l'ordre des filtres n'a pas été effectuée et représente une faille importante de notre approche. Par le raisonnement décrit ci-haut, on peut affirmer que changer l'ordre des filtres permettrait d'améliorer la sensibilité du modèle : le filtre le moins sensible étant mis à la fin modifierait l'ensemble de données, et à l'entrée, et à la sortie de ce même filtre.

6.5.2 Justification de l'approche

On peut se questionner sur la légitimité d'être le plus spécifique possible à chacune des étapes de la méthode présentée. En effet, en agissant ainsi, on peut éliminer des candidats qui sont en fait de vrais snoRNAs non encore identifiés. Il est donc utile de rappeler au lecteur deux éléments directeurs de la recherche présentée :

- On souhaite être le plus spécifique possible tout en gardant une bonne sensibilité (> 0.7, voir section 2.9).
- On veut découvrir de nouveaux snoRNAs, et, étant donné que notre approche ne se veut pas automatique, on est prêt à perdre en sensibilité.

La méthode développée ne permet pas d'avoir une mesure de qualité entre les FP, elle permet seulement d'éliminer un ensemble de candidats qui se révèle moins semblable aux vrais snoRNAs. Dans cette optique, être sévère dans les critères de recherche (augmenter la spécificité) permet d'éliminer plus de FP.

Dans la suite de cette discussion, j'essaierai de justifier la première prémisse présentée ci-haut pour chacune des étapes du modèle. L'étape SVM n'est pas abordée puisque l'ajustement de ses paramètres est indépendant du modèle.

Étape de recherche de snoRNA

Pour cette étape de recherche, si on autorise une erreur sur la boîte C, le nombre de FP passe de 2686 à 6800. Par contre, le nombre de TP retrouvé augmente aussi : il passe de 31 à 40. Avec le modèle présenté (voir figure 5.1), et en permettant cette erreur, il est impossible de descendre sous la barre des 100 FP avant l'étape SVM. Cette avenue a donc été écartée de la recherche à cause du nombre élevé de candidats.

À la lumière de la discussion de la section précédente, il est clair que l'ordre des filtres pourrait avoir un impact sur la qualité du modèle. De plus, comme il a été mentionné en section 2.5, près de la moitié des snoRNAs répertoriés ne respectent pas le consensus de la boîte C. En conséquence, il semble injustifié d'être le plus spécifique possible car le modèle produit ne reflète pas la famille des snoRNAs.

Étape de recherche de tetraloop

Au contraire de l'étape de recherche de snoRNAs, l'étape tetraloop a été analysée exhaustivement (voir figure 5.6). On peut donc justifier la sévérité des paramètres employés puisque, comme le montre la figure, le maximum de TP qu'on peut récupérer est 26 / 31. De plus, des 31 vrais snoRNAs de départ, seulement 23 possèdent une tetraloop clivée expérimentalement, 18 étant exactement celles retrouvées par ma méthode. On peut donc conclure que les valeurs de paramètres choisis permettent un modèle qui permet de bien calquer la classe des tetraloops clivées expérimentalement.

Étape de validation par homologie de séquences

Le seuil de E-value reflète le degré de conservation de notre ensemble de TP snoRNAs. Plus on est sévère avec ce paramètre, plus on exerce de pression vers les séquences ayant une bonne conservation de séquence, ce qui est parfaitement valide pour une famille fonctionnelle d'ARN.

6.6 Conclusion

Une analyse exhaustive des 63 candidats restants (31 TP et 32 FP) nous permet d'affirmer que 3 candidats sont intéressants en lien avec nos 31 TP snoRNAs. Le plus intéressant est psnr624. Les critères de sélection de cette analyse étaient le résultat donné par snoSCAN, par DRSD, et par une analyse de la concentration cellulaire des transcrits. L'analyse par snoSCAN ayant été moins fructueuse, seulement les deux autres analyses ont été prises en compte pour le choix des candidats finaux. Psnr624 est le candidat par excellence parce qu'en plus de montrer une différence de concentration des transcrits et de posséder une tetraloop ayant un score DRSD faible (0.20), la taille du transcrit snoRNA est similaire à la taille requise. Parmi les 23 snoRNAs dont la tetraloop est clivée expérimentalement, 20 de ceux-ci répondent à ces critères.

Il serait approprié de modifier les paramètres pour l'étape de recherche de snoRNAs et de modifier l'ordre des filtres. Je suis convaincu qu'il est possible d'améliorer la sensibilité du modèle et d'obtenir tout de même un ensemble final de candidats inférieur à 100 avant l'étape SVM.

CHAPITRE 7

CONCLUSION

Les snoRNAs de type C/D sont des ARN non-codants ayant des caractéristiques bien définies dans la littérature. Certaines observations nous permettent de penser qu'il resterait environ une dizaine de ces molécules non répertoriées chez la levure *Saccharomyces cerevisiae*. La séquence des snoRNAs présentant très peu de conservation structurale ou séquentielle, il est très difficile de définir un consensus pour cette famille d'ARN, et les méthodes de recherche existantes donnent nécessairement des résultats partiels. La méthode bio-informatique qui a permis de découvrir le plus de snoRNAs de type C/D est l'algorithme snoSCAN [51]. Cependant snoSCAN est limité à la découverte des snoRNAs méthylant des nucléotides de l'ARNr.

Dans ce mémoire, notre objectif est de découvrir de nouveaux snoRNAs, n'ayant pas nécessairement cette fonction de méthylation de l'ARN ribosomal. Pour ce faire, nous avons utilisé une caractéristique commune aux snoRNAs, non prise en compte par les méthodes existantes. Il s'agit de la présence, en amont ou en aval de la séquence du snoRNA, d'un motif tetraloop représentant le site de clivage de l'enzyme RNT1P. Afin de se servir de cette information pour retrouver de nouveaux snoRNAs, il a été nécessaire tout d'abord de caractériser ce motif, en alignant les séquences connues (clivées expérimentalement), et en utilisant la statistique d'information mutuelle pour retrouver les contraintes d'appariement et d'interdépendance entre les positions du tetraloop.

Nous nous limitons dans ce mémoire à la recherche de nouveaux snoRNAs dans le génome de la levure *Saccharomyces cerevisiae*. Notre nouvelle méthode se base sur la recherche de séquences et d'hélices conservées (séquence génomique du snoRNA, ainsi que tetraloop à une distance donnée en amont ou en aval), sur la validation des candidats par génomique comparative, et sur un algorithme de classification efficace. Les contraintes et les paramètres ont été définis de façon à être très restrictifs en ce qui a trait au nombre de candidats retrouvés, et à fournir une confiance maximale dans la pertinence des candidats finaux obtenus. L'objectif n'étant pas de développer un outil automatisé

permettant de retrouver tous les snoRNAs, mais bien une méthode supplémentaire permettant de découvrir de nouveaux snoRNAs, on s'attendait à ce que certains TP snoRNAs ne soient pas reconnus, bien que cet état de fait ne soit pas souhaitable. Comme il a été discuté en section 6.5, une modification dans l'ordre des étapes pourrait mener à une amélioration de cette situation. Cependant, un élément permettant de renforcer la confiance que l'on a en notre méthode est que parmi les 18 snoRNAs répertoriés finaux, 16 font partie des 23 TP pour lesquels il existe une tetraloop clivée expérimentalement. La méthode proposée retrouve des snoRNAs d'une façon qui est reliée au mécanisme de maturation, plutôt qu'à la fonction de ces molécules.

Les résultats finaux font apparaître 32 séquences tetraloops qui seraient des substrats clivable par l'enzyme RNT1P. Ces même séquences, possédant un snoRNA de type C/D dans leurs voisinages, pourraient être les motifs de clivage nécessaires à la libération de ceux-ci. Par contre, l'étude de nos TP snoRNAs nous indique que seulement 3 de ces 32 prédictions ont un profil différentiel de concentration de transcrit. Une validation expérimentale permettrait de confirmer ces prédictions.

Les candidats retrouvés ne font pas partie des snoRNAs méthylant des nucléotides de l'ARNr. Si les candidats se révèlent validés en laboratoire, alors on obtiendra une caractéristique novatrice permettant de cibler des snoRNAs de type C/D inconnus. Un logiciel utilisant cette caractéristique devrait être plus général que snoSCAN [51], utilisant le motif ASE (portion complémentaire à l'ARNr), puisque l'on n'est pas obligé de spécifier le type d'ARN sur lequel les nucléotides méthylés se retrouvent. La méthode proposée retourne des candidats que snoSCAN ne parvient pas à cerner.

L'équipe de S. Abou Elela a confirmé certaines de nos prédictions. Des résultats préliminaires avaient été envoyés et sur les 7 candidats testés en laboratoires, 4 ont montré une différence de concentration de transcrit entre une cellule de *Saccharomyces cerevisiae* sauvage et une cellule Δ RNT (cellule dont le gène codant l'enzyme RNT1P est non-fonctionnel), suggérant que 4 de ces transcrits sont traités par l'enzyme.

La méthode pourrait s'appliquer à des génomes autres que celui de *Saccharomyces cerevisiae*, puisque les motifs recherchés pour le snoRNA sont conservés par exemple chez l'humain [51]. Par contre, il ne nous est pas possible de garantir que le tetraloop

serait similaire en dehors du groupe des Hémiascomycètes. Une étude des mécanismes génétiques contrôlant l'expression des snoRNAs chez l'humain serait appropriée afin de vérifier la validité du modèle de recherche.

Le mécanisme de clivage existe aussi pour la classe de snoRNAs H/ACA, c'est donc une contrainte qui serait utile à vérifier pour ces molécules.

BIBLIOGRAPHIE

[1] Balakin A., Smith L., and Fournier M.J. The RNA World of the Nucleolus : Two Major Families of Small RNAs Defined by Different Box Elements with Related Functions. *Cell*, 86 :823–834, 1996.

[2] Markov A.A. An example of statistical investigation in the text of "Eugene onyegin" illustrating coupling of "tests" in chains. *In Proceedings of Academic Scienctific St. Petersburg*, 6 :153–162, 1913.

[3] Hamilton A.J. and Baulcombe D.C. A species of small antisense RNA in posttranscriptional gene silencing in plants. *Science*, 286(5441) :950–952, 1999.

[4] Lowe T. and Eddy S. http ://www.cse.ucsc.edu/ lowe/. Site web.

[5] Zeng Y. and Cullen B.R. Recognition and cleavage of primary microRNA transcripts. *Methods Mol Biol.*, 342 :49–56, 2006.

[6] Chang C. and Lin C. *LIBSVM : a library for support vector machines*, 2001.

[7] Gautheret D., Major F., and Cedergren R. Pattern searching/alignment with RNA primary and secondary structures : an effective descriptor for tRNA. *Comput. Appl. Biosci.*, 6 :325–331, 1990.

[8] Ge D. and Elela S.A. RNase III-mediated silencing of a glucose-dependent repressor in yeast. *Curr Biol.*, 15, 2005.

[9] Knuth D.E., Morris J.H., and Pratt V.R. Fast Pattern Matching in Strings. *SIAM Journal on Computing*, 6(2) :323–350, 1977.

[10] Bartel D.P. Micrornas : genomics, biogenesis, mechanism, and function. *Cell*, 116 :281–297, 2004.

[11] Altschul S.F. *et al.*. Basic local alignment search tool. *J Mol Biol*, 215 :403–410, 1990.

[12] Beata E. *et al.*. A small nucleolar guide RNA functions both in 2'-O-ribose methylation and pseudouridylation of the U5 spliceosomal RNA. *The EMBO Journal*, 20 :541–551, 2001.

[13] Cliften P. *et al.*. Surveying *Saccharomyces* genomes to identify functional elements by comparative DNA sequence analysis. *Genome Research*, 11 :1175–1186, 2001.

[14] David L. *et al.*. A high resolution map for transcription in the yeast genome. *PNAS*, 103(14) :5320–5225, 2006.

[15] Edvardsson S. *et al.*. A search for H/ACA snoRNAs in yeast using MFE secondary structure prediction. *Bioinformatics*, 19(7) :865–873, 2003.

[16] El-Mabrouk N. *et al.*. Approximate matching of structured motifs in DNA sequences. *Journal of Bioinformatics and Computational Biology*, 3(2) :317–342, 2005.

[17] Eo H.S. *et al.*. A combined approach for locating box H/ACA snoRNAs in the human genome. *Mol. Cells.*, 20(1) :35–42, 2005.

[18] Ghazal G. *et al.*. Genome-wide prediction and analysis of yeast RNase III-dependent snoRNA processing signals. *Molecular and cellular biology*, 25(8) :2981–2994, April 2005.

[19] Guldener U. *et al.*. CYGD : the Comprehensive Yeast Genome Database. *Nucleic Acids Research*, 33, 2005.

[20] Hofacker I.L. *et al.*. Fast folding and comparison of RNA secondary structures. *Monatshefte für Chemie*, 125 :167–188, 1994.

[21] Lamontagne B. *et al.*. Sequence dependence of substrate recognition and cleavage by yeast Rnase III. *Journal of molecular biology*, 327 :985–1000, 2003.

[22] Lamontagne B. *et al.*. Molecular requirements for duplex recognition and cleavage by eukaryotic Rnase III : discovery of an RNA- dependant DNA cleavage activity of yeast Rnt1p. *Journal of molecular biology*, 338 :401–418, 2004.

[23] Macke T.J. *et al.*. RNAMotif, an RNA secondary structure definition and search algorithm. *Nucleic Acids Research*, 29(22) :4724–4735, 2001.

[24] Massenet S. *et al.*. *Modification and Editing of RNA*, chapter Post-transcriptional modifications in the U small nuclear RNAs, pages 201–227. ASM Press, 1998.

[25] Omer A.D. *et al.*. Homologs of small nucleolar RNAs in Archaea. *Science*, 288 :17–522, 2000.

[26] Wu H. *et al.*. Structural basis for recognition of the AGNN tetraloop RNA fold by the double stranded RNA-binding domain of RNT1p RNAseIII. *PNAS*, 101 :8307–8312, 2004.

[27] Xue *et al.*. Classification of real and pseudo microrna precursors using local structure-sequence features and support vector machine. *BMC Bioinformatics*, 6, 2005.

[28] Lisacek F., Diaz Y., and Michel F. Automatic identification of group I introns cores in genomic DNA sequences. *Journal of molecular biology*, 235 :1206–1217, 1997.

[29] Chanfreau G. Conservation of Rnase III processing pathways and specificity in hemiascomycetes. *Eukaryotic Cell*, 2 :901–909, 2003.

[30] Chanfreau G., Legrain P., and Jacquier A. Yeast RNase III as a key processing enzyme in small nucleolar RNAs metabolism. *Journal of Molecular Biology*, 284 :975–988, 1998.

[31] Mauri G. and Pavesi G. Algorithms for pattern matching and discovery in RNA secondary structure. *Theoretical Computer Science*, 335 :29–51, 2005.

[32] Rebane A.and Roomere H. and Metspalu A. Locations of several novel 2'-O-methylated nucleotides in human 28S rRNA. *BMC Molecular Biology*, 3(1), 2002.

[33] Hartley H.O. Maximum likelihood estimation from incomplete data. *Biometrics*, 14 :174–194, 1958.

[34] Ni J., Tien A.L., and Fournier M.J. Small Nucleolar RNAs Direct Site-Specific Synthesis of Pseudouridine in Ribosomal RNA. *Cell*, 89 :565–573, 1997.

[35] Nowakowski J. *Oxford handbook of nucleic acid structure*, chapter RNA structure in solution, page 568. Oxford, 1999.

[36] Emerson J.J., Kaessmann H., and Betrán E. Extensive gene traffic on the mammalian x chromosome. *Science*, 303(5657) :537–540, 2004.

[37] Boyer R.S. and Moore J.S. A fast string searching algorithm. *Communications of the ACM*, 20 :762–772, 1977.

[38] Yeom K.H., Lee Y., Han J., Suh M.R., and Kim V.N. Characterization of DGCR8/Pasha, the essential cofactor for Drosha in primary miRNA processing. *Nucleic acids research*, 34 :4622–4629, 2006.

[39] Smith l., Andersen K.B., Hovgaard L., and Jaroszewski J.W. Rational selection of antisense oligonucleotide sequences. *Eur J Pharm Sci*, 11 :191–198, 2000.

[40] Zuker M., Mathews D.H., and Turner D.H. *Algorithms and Thermodynamics for RNA Secondary Structure Prediction : A Practical Guide*. RNA Biochemistry and Biotechnology, 1999.

[41] Waldrop M.M. Catalytic RNA wins chemistry Nobel. *Science*, 246(4928) :325, 1989.

[42] El-Mabrouk N. and Lisacek F. Very fast identification of RNA motifs in genomic DNA. Application to tRNA search in the yeast genome. *Journal of Molecular Biology*, 264 :46–55, 1996.

[43] Gregory R.I., Chendrimada T.P., and Shiekhattar R. MicroRNA biogenesis : isolation and characterization of the microprocessor complex. *Methods Mol Biol.*, 342 :33–47, 2006.

[44] Eddy S. and Durbin R. RNA sequence analysis using covariance models. *Nucleic acids research*, 22 :2079–2088, 1994.

[45] Eddy S. and Lowe T. tRNAscan-SE : a program for improved detection of transfer RNA genes in genomic sequence. *Nucleic Acids Research*, 25(5) :955–964, 1997.

[46] Geisser S. The Predictive Sample Reuse Method with Applications. *Journal of the American Statistical Association*, 70(350) :320–328, 1975.

[47] Steinberg S. and Loudovitch A. A role for the bulged nucleotide 47 in the facilitation of tertiary interactions in the tRNA structure. *RNA*, 2 :84–87, 1996.

[48] Chang T., Huang H., Chuang T., Shien D., and Horng J. RNAMST : efficient and flexible approach for identifying RNA structural homologs. *Nucleic acids research*, 34 :W423–W428, 2006.

[49] Kin T. and Asai K. Marginalized kernels for RNA sequence data analysis. *Genome Informatics*, 13 :112–122, 2002.

[50] Smith T.F. and Waterman M.S. Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, 147 :195–197, 1981.

[51] Lowe T.M. and Eddy S.R. A computational screen for methylation guide snoRNAs in yeast. *Science*, 283 :1168–1171, 1999.

[52] Aho A. V. and Corasick M. J. Efficient string matching : an aid to bibliographic search. *CACM*, 18(6) :333–340, 1975.

[53] Gilbert W. The RNA world. *Nature*, 319 :618, 1986.

[54] Luo Y. and Li S. Genome-wide analyses of retrogenes derived from the human box H/ACA snoRNAs. *Nucleic Acids Research*, 35(2) :559–571, 2006.

[55] Yao Y.Y. *Information-theoretic measures for knowledge discovery and data mining, in Entropy Measures, Maximum Entropy Principle and Emerging Applications*, pages 115–136. Springer, 2003.

Annexe I

I.1 Vrais snoRNAs

Dans la figure 5.1, les valeurs sur les flèches indiquent le nombre de candidats restants. Un ensemble a été utilisé afin d'évaluer la sensibilité de la méthode, il s'agit d'un ensemble de vrais snoRNAs (TP), qui sont répertoriés sur le site CYGD [19]. Ceux-ci, au nombre de 43 et présentés dans la table I.1, passent au travers des différents filtres de la méthode et sont en quelque sorte une mesure des portions plus restrictive de l'approche.

I.2 Ensemble d'entraînement pour le SVM

Pour entraîner le SVM, il est nécessaire d'avoir en main plusieurs exemples et contre-exemples. Un exemple, dans le cas qui nous intéresse, est un motif tetraloop respectant les contraintes définies en section 5.2.2. Les substrats clivés expérimentalement testés par l'équipe de Sherif sont nos exemples. Pour les contre-exemples, nous avons utilisé RNAMOTIF (voir section 4.4) afin de trouver des tige-boucles avec une hélice de longueur 6 et une boucle variant de 3 à 6nt sur le génome de *Saccharomyces cerevisiae*. Dans le cas où une occurrence retournée par RNAMOTIF était une tetraloop, on a subséquemment rejeté les motifs ayant une boucle de type NGNN. Comme on peut le constater dans la table I.2, il existe un déséquilibre entre le nombre d'exemples et de contre-exemples. Le rapport est de 7.6 :1 en faveur des contre-exemples. L'option -w de libSVM [6] permet de contre-balancer cet effet en ajoutant plus de pénalité à une classe particulière. On a donc ajouté une pénalité 7 fois supérieure pour les exemples qui se retrouvent mal classifiés.

I.3 SVM

SVM est l'acronyme pour Support Vector Machine. C'est un algorithme d'apprentissage permettant de répondre à plusieurs problèmes dont, la classification binaire. Le

Nom	chr	début	fin	longueur
U18	01	142371	142472	101
Snr56	02	88188	88275	87
Snr65	03	177179	177278	99
Snr13	04	1402910	1403033	123
Snr63	04	323471	323217	254
Snr47	04	541695	541597	98
Snr4	05	424683	424858	175
Snr52	05	431216	431125	91
Snr67	05	61352	61433	81
Snr53	05	61699	61789	90
Snr39	07	365252	365164	88
Snr39b	07	366470	366375	95
Snr48	07	609586	609698	112
Snr71	08	411229	411318	89
Snr68	09	97111	97246	135
Snr128	10	139611	139484	127
Snr190	10	139868	139679	189
Snr60	10	348933	348830	103
Snr38	11	282830	282924	94
Snr69	11	364419	364519	100
Snr64	11	38812	38912	100
Snr61	12	794575	794486	89
Snr55	12	794794	794697	97
Snr57	12	795024	794937	87
Snr54	13	163620	163535	85
SnRNAZ7	13	297506	297589	83
SnRNAZ6	13	297724	297832	108
SnRNAZ3	13	298306	298406	100
SnRNAZ2	13	298554	298644	90
U24	13	500071	499983	88
Snr19	14	230258	230104	154
Snr66	14	586088	586173	85
Snr40	14	89208	89298	90
Snr58	15	136182	136087	95
Snr50	15	259489	259578	89
Snr62	15	409863	409764	99
Snr17a	15	780107	780596	489
Snr59	16	173826	173903	77
Snr17b	16	281516	281055	461
Snr51	16	718802	718696	106
Snr70	16	719046	718883	163
Snr41	16	719247	719153	94
Snr45	16	821725	821896	171

Tableau I.1 – Description des vrais snoRNAs utilisés provenant du site web CYGD [19]. La première colonne correspond au nom donné pour la molécule, la seconde au numéro du chromosome, vient ensuite les bornes de la molécule dans le génome, puis la longueur de ces éléments.

	triloop	tetraloop	pentaloop	hexaloop
exemple	0	44	0	0
contre-exemple	81	43	146	22

Tableau I.2 – Quantités respectives des différentes tiges-boucles utilisées lors de l'entraînement du SVM.

problème s'énonce de la façon suivante : étant donné N échantillons d'apprentissage, distribués en 2 classes, est-il possible de trouver une séparation (possiblement non-linéaire) au travers de leurs caractéristiques qui reflète la véritable séparation si on avait tous les motifs des 2 classes à notre disposition. SVM est un algorithme efficace et fiable puisqu'il essaie de maximiser la marge, c'est-à-dire la région autour de la frontière de décision, de façon à bien séparer l'interface de classe, et il n'est pas sensible à l'ordre dans lequel les données sont présentées. On appelle cet algorithme de cette façon puisque seulement certains points définissent le modèle appris, ce sont les Support Vectors (SV), ou plus simplement les échantillons qui se retrouvent sur la marge, donc qui vérifient l'équation suivante :

$$y_i(x_i + b) = 1$$

Où y_i correspond à l'étiquette de l'échantillon, x_i correspond à l'échantillon, et b la marge.

Lorsque le modèle est construit et qu'un nouvel échantillon est testé, la classe de celui-ci est assignée grâce à la formule suivante :

$$f(x) = sgn(\sum_i^N y_i \alpha_i K(x, x_i) + b)$$

Donc le point de test x est comparé (via la fonction de noyaux K) à tous les SVs du modèle (x_i), et le signe de cette évaluation sert à donner l'appartenance à la classe.

Le SVM possède un hyper-paramètre, dénoté C, ou fonction de pénalité, qui joue sur le modèle de la façon suivante :

$0 \leq \alpha_i \leq C$

Cette fonction permet de donner des poids très importants à certains SVs du modèle, c'est ce facteur qui contrôle le sur-apprentissage. Plus le C est élevé, plus le SVM va avoir tendance à complexifier la frontière de décision, et donc en général une grande valeur de C est synonyme de sur-apprentissage.

L'autre hyper-paramètre à régler concerne la fonction de noyaux K, soit l'écart-type de la gaussienne, ou plus simplement qui est définit par $\gamma = \frac{1}{\sigma^2}$. Plus la valeur de γ est élevé (plus l'écart type est faible), et donc la courbe gaussienne deviendra de plus en plus étroite, la relation de voisinage entre deux points tombant à zéro très rapidement. La fonction de noyau permet d'approximer des fonctions non linéaires en faisant une correspondance de dimensionalité dans un espace plus grand, par produit scalaire. On peut ainsi passer dans un espace non-linéaire avec le même jeu de données.

Lorsque l'on entraîne le modèle SVM, on cherche le vecteur des α qui détermine les SVs. Pour ce faire, on doit résoudre le problème dual suivant :

$LD \equiv \sum_i^N - \frac{1}{2}[\sum_i^N \sum_j^N \alpha_i \alpha_j y_i y_j K(x_i, x_j)]$

sujet à : $0 \leq \alpha_i \leq C$, $\sum_i^N \alpha_i y_i = 1$ et :

$LP \equiv \frac{1}{2} w w^t + C \sum_i^N \varepsilon_i - \sum_i^N \alpha_i (y_i (x_i w + b) - 1 + \varepsilon_i) - \sum_i^N w_i \varepsilon_i$

où LD = Langrangian Dual, LP = Langrangian primal, w représente le vecteur normal à la frontière de décision, et correspond à une variable de pénalité associée aux différents points de l'ensemble d'entraînement. N étant la cardinalité de l'ensemble d'entraînement. Il s'agit à la fois de maximiser LD tout en minimisant LP. Cette procédure est faite par programmation quadratique.