UNIVERSITÉ DU QUÉBEC À MONTRÉAL

PRÉDICTION DE STRUCTURE SECONDAIRE D'ARNS AVEC PSEUDO-NŒUDS ET DE SES MOTIFS STRUCTURAUX GRÂCE À LA PROGRAMMATION ENTIÈRE

MÉMOIRE

PRÉSENTÉ

COMME EXIGENCE PARTIELLE

DE LA MAÎTRISE EN INFORMATIQUES

PAR

GABRIEL LOYER

JUILLET 2023

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

RÉSUMÉ

La prédiction d'une structure secondaire d'ARN contenant des pseudo-nœuds reste un défi dans les modèles thermodynamiques. L'énergie des motifs 3D locaux rejoignant les tiges canoniques est approximée. De plus, même si les interactions de pseudo-nœuds sont nombreuses et importantes, elles sont souvent ignorées en raison de la complexité supplémentaire et de la précision réduite en utilisant des approches de programmation dynamique standard pour calculer la structure la plus stable. Parallèlement, il est devenu de plus en plus évident ces dernières années que les motifs structurels dans les boucles, composés d'interactions non canoniques, sont essentiels pour la forme finale de la molécule permettant ses multiples fonctions. Notre capacité à prédire des structures 3D précises est également limitée lorsqu'il s'agit de l'organisation du grand réseau complexe d'interactions qui se forment à l'intérieur de ces boucles.

Le logiciel RNA-MoIP (RNA Motifs over Integer Programming) a précédemment été développé pour concilier la structure secondaire de l'ARN et les informations sur les motifs 3D locaux disponibles dans les bases de données. J'ai approfondi notre modèle afin de prédire simultanément les paires de bases canoniques (incluant des pseudo-nœuds) à partir des matrices de probabilité des paires de bases. L'utilisation de décompositions de structures contenant des pseudo-nœuds en de multiples structures sans pseudo-nœud permet d'utiliser un modèle thermodynamique pour approximer les probabilités de paires générales, qui est ensuite guidé par l'insertion de motifs. J'ai également ajouté la possibilité de donner une prédiction de structure secondaire sans avoir besoin d'une structure de base comme référence en entrée. La prédiction est ensuite raffinée par itérations, avec la structure prédite comme contrainte, jusqu'à ce que la convergence soit atteinte. J'ai ensuite évalué notre nouvelle méthode sur une base non redondante de toutes les structures d'ARN de moins de 150 nucléotides. Je montre que la prédiction conjointe de la structure des paires de bases canoniques et des motifs conservés locaux (i) améliore le ratio des interactions bien prédites dans la structure secondaire qui contienne des pseudo-nœuds, et (ii) diminue le ratio de paires de bases erronées.

Le code source du cadriciel ainsi que la base de données et les résultats sont disponibles sur le répertoire suivant : `https://gitlab.info.uqam.ca/cbe/RNAMoIP`. Un serveur web est également accessible sous l'adresse `https://rnamoip.cbe.uqam.ca` afin d'effectuer une prédiction sous une interface utilisateur.

CHAPITRE I

INTRODUCTION

INTRODUCTION

L'essor de la thérapeutique par ARN (Wang *et al.*, 2020; Yu *et al.*, 2020) est dû aux avancées techniques et informatiques dans notre compréhension du paradigme séquence-structure-fonction de l'ARN. Si la prédiction de la structure de l'ARN tous atomes confondus à partir de la séquence reste encore un défi (Miao *et al.*, 2020), de nombreuses approches différentes ont permis d'obtenir des résultats intéressants dans différentes facettes du problème.

Tirant avantage du repliement hiérarchique de l'ARN, la structure secondaire composée des tiges fortes composées de paires de bases canoniques et Wobble se forment en premier (Tinoco et Bustamante, 1999). De nombreuses approches théoriques efficaces basées sur le modèle du plus proche voisin ont été développées pour prédire cette structure secondaire. Néanmoins, les modèles les plus précis et les plus réalisables, comme dans la librairie ViennaRNA (Lorenz *et al.*, 2011) ou RNAstructure (Reuter et Mathews, 2010), supposent qu'il n'y a pas d'interactions croisées, pas de pseudo-nœud, car cette hypothèse ajoute de la complexité et diminue la précision des paramètres thermodynamiques, ce qui rend leur utilisation souvent peu pratique. Pourtant, les pseudo-nœuds sont abondants et importants. Prédire une structure secondaire précise avec ceux-ci serait inestimable pour les principaux outils de reconstruction 3D qui reposent sur ce type de structure secondaire (Watkins *et al.*, 2020).

Les structures d'ARNs ne se composent pas seulement d'interactions canoniques ou Wobble. La classification de Leontis-Westhof (LF) (Leontis et Westhof, 2001) définit 12 types de paires de bases possibles, entre n'importe quels nucléotides. Lors

de la description des boucles entre les tiges rigides en utilisant ces interactions non canoniques, différentes méthodes ont montré que des sous-structures conservées sont présentes et importantes (Leontis *et al.*, 2006; Reinharz *et al.*, 2018).

Les travaux antérieurs ont utilisé une base de données de motifs structurels conservés pour sélectionner une structure secondaire optimale et faciliter la reconstruction 3D de tous atomes (Reinharz *et al.*, 2012; Yao *et al.*, 2017). Dans ce travail, j'étends ce cadriciel de programmation entière pour inclure la prédiction simultanée de la structure secondaire avec des pseudo-nœuds et l'insertion de motifs structurels.

Dans cette introduction, j'introduis les matières nécessaires à la bonne compréhension du sujet, par rapport à des notions biologiques et d'algorithmique. Par la suite, la méthodologie utilisée sera expliquée plus en détail. Ensuite, les résultats des expériences seront démontrés et analysés, pour enfin terminer sur la conclusion de ces travaux.

1.1 ARN

Dans le domaine de la biologie, l'acide ribonucléique, ou simplement ARN, est une des trois composantes primaires du dogme central de la biologie moléculaire. Cette macromolécule, transcrite depuis les informations de l'ADN, joue un rôle primordial dans plusieurs activités de nombreux organismes : régulation de gènes, traductions en protéines, transmission ou interprétation de message, etc. On retrouve aussi certains virus qui ont leurs informations génétiques encodées sous forme d'ARN.

La principale caractéristique qui différencie l'ARN de l'ADN se renferme dans leur composition chimique : le nucléotide T, la Thymine, est transcrit dans l'ARN vers le nucléotide U, l'Uracile. Du côté de l'ADN, la thymine forme des liens chimiques avec l'Adénine (A) de même longueur que l'autre lien présent dans l'ADN, la Cytosine et la Guanine (C-G). Cela permet à la structure de l'ADN de se replier en fameuses doubles hélices de façon stable, puisque les atomes du squelette vont parfaitement se superposer, et les atomes des nucléotides vont occuper le même espace. Par contre, du côté de l'ARN, la Guanine à une tendance à former des liens non seulement avec la Cytosine, mais également l'Uracile. Les liens chimiques que l'on retrouve dans la structure demeurent isostériques (géométriquement égaux), à l'exception du lien G-U. C'est ainsi que ce bris de symétrie fait en sorte que la structure de l'ARN se replie de façon différente que de la simple hélice de l'ADN. La structure se replie sur elle-même, dans un état stable, de façon à trouver un état de repos où l'énergie libre est au minimum. Cette forme finale dépend des interactions chimiques qui se créer dans la structure, qui dépend de facto de la séquence de nucléotides. On définit l'ensemble des interactions chimiques d'une séquence comme étant les paires de bases de nucléotides, ce qui constitue la structure secondaire de l'ARN.

Un des défis principaux de la bio-informatique est de tenter de prédire les interactions chimiques potentielles d'une séquence de nucléotides donnés. Cette opération est particulièrement utile lors d'analyse de nouvelles séquences d'ARNs dont lesquelles la structure n'est pas encore connue. Puisque les techniques pour séquencer les nucléotides sont beaucoup plus accessibles que celles pour identifier la structure, cette approche de prédiction devient intéressante. En effet, même si elles existent, les techniques pour identifier la structure reste très coûteuses en temps et en matériel.

Une des hypothèses fondamentales formulées à ce jour repose dicte que la structure apporte une grande influence sur la fonction de l'ARN. Ainsi, si l'on connaît sa structure secondaire, on peut déduire les capacités de la molécule. À noter que la structure secondaire constitue une étape importante pour établir une prédiction adéquate de la structure en trois dimensions.

Toutefois, les paires canoniques ne sont pas les seules interactions qui constituent une structure d'ARN. Dans le tableau 1.1, on définit les 12 types d'interactions non canoniques possibles de la nomenclature Leontis-Westhof (Leontis et Westhof, 2001). Cette liste est basée sur la combinaison des trois côtés possibles du lien d'interaction, abstraction géométrique d'un triangle. Ainsi, l'interaction peut avoir lieu du côté Watson–Crick, Hoogsteen ou du côté du sucre, en plus d'être orienté avec les liaisons osidiques (cis) ou non (trans) (voir Fig. 1.3).

Ces interactions sont particulièrement importantes pour la structure en trois dimensions. Démontrer dans plusieurs études, comme dans le cas du contact A-Minor (Reinharz *et al.*, 2018), ces sous-structures sont conservées dans plusieurs séquences, tout en se repliant de la même façon. Ceci permet donc d'identifier les potentiels réseaux afin d'obtenir une reconstruction spatiale de la structure plus rapidement (Watkins *et al.*, 2020). Toutefois, même à partir de nombreux para-

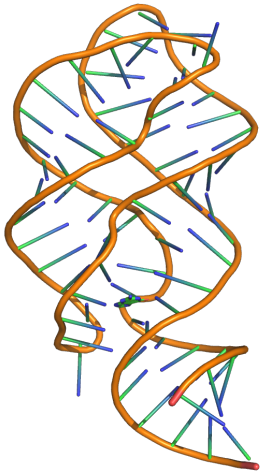


FIGURE 1.1 – **Représentation 3D d'un PDB**. Le riboswitch 1Y27, où sa structure joue un rôle important dans la liaison avec d'autres petits ligands pour l'expression de gènes.

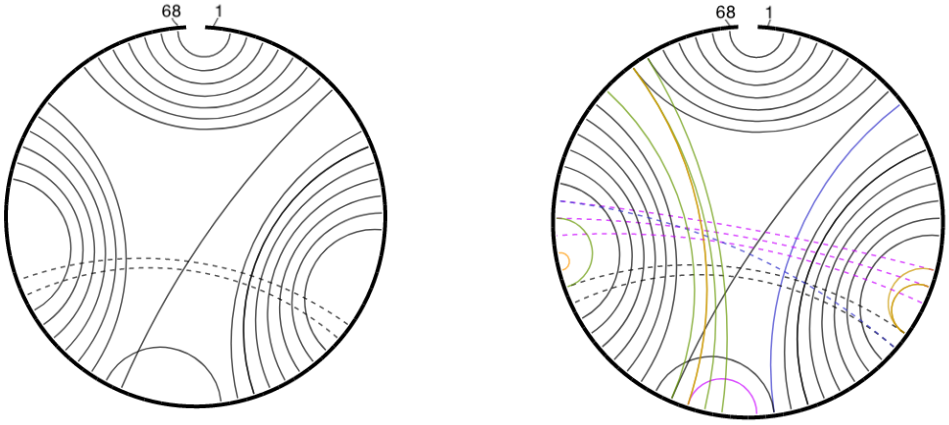


FIGURE 1.2 – **Représentation des interactions de 1Y27**. À gauche, les interactions canoniques, incluant les pseudo-nœuds représentées en pointillés. À droite, les interactions non canoniques sont également représentées sous différentes couleurs, dépendamment de leur type(BGSU RNA, 2022). À noter que les deux paires de bases isolées sont exclues de la structure secondaire, puisque le modèle énergétique requiert un minimum de deux paires de bases empilées pour considérer leur énergie.

mètres et informations reliées à la séquence, il est difficile de prédire empiriquement la présence ou non de ces motifs, malgré leur importance structurale (Gianfrotta

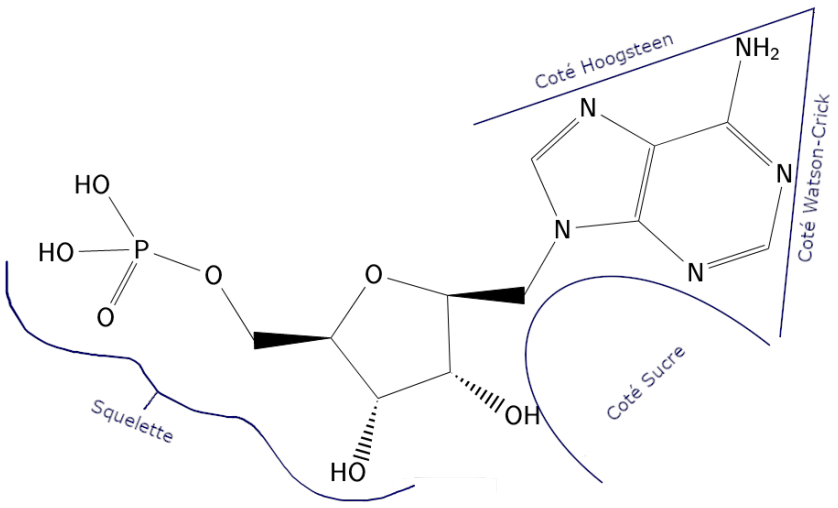


FIGURE 1.3 – **Adénine**. Les trois côtés de la liaison de l'Adénine, représentés sous forme de triangle. Tous les nucléotides sont toujours reliés au squelette.

Non-Canonical Interactions			
	cWW		cHH
	tWW		tHH
	cWH		cHS
	tWH		tHS
	cWS		cSS
	tWS		tSS

TABLEAU 1.1 – **Différents types d'interactions non canoniques**. Tous les interactions possibles entre les côtés Watson–Crick (W), Hoogsteen (H) ou sucre (S), de façon cis (c) ou trans (t).

et al., 2022).

1.2 Algorithmie

La méthode de résolution principale pour la problématique présentée fut par une approche plus mathématique, par la programmation entière (Integer Programming). Cette approche focalise sur l'énumération de variables, contraintes par un système d'équations. À partir de ce système, l'objectif est de trouver les valeurs des variables qui optimiseraient une fonction préalablement définie comme la fonction objective. Cette optimisation peut se faire par une maximisation ou bien une minimisation pour la valeur retournée par la fonction objective.

$$\begin{aligned} \text{Maximise} : \sum_{i \in I} p_i \cdot x_i \\ \text{Sujet à} : \sum_{i \in I} w_i \cdot x_i \leq c \end{aligned} \tag{1.1}$$

1.1 – **Exemple de système d'équations pour la représentation du problème du sac à dos**. Un poids et un prix sont associés à plusieurs objets, et on désire maximiser le profit par rapport à un poids maximal c. Dans cet exemple, w_i et p_i correspondent au poids et à la valeur de l'objet i, et la variable x_i indique si l'objet est choisi ou non.

Un avantage de la programmation entière est qu'elle permet de représenter des problèmes complexes en un ensemble d'équations mathématiques. Toutefois, la résolution d'un système d'équations à variables entières est fondamentalement très difficile. À la base, la complexité du problème a été démontrée à maintes reprises comme étant NP-Complet (Papadimitriou, 1981). Néanmoins, puisque sa représentation est universelle, il est donc possible d'utiliser des logiciels pour résoudre ces systèmes d'équations de façon efficace et optimale. Ainsi, la complexité de résolution du problème est donc transférée à un parti tiers, et on peut se concentrer à bien identifier la problématique ainsi qu'à la façon de la définir. Même si le problème est NP-Complet, ces solveurs ont réussi à implémenter différentes techniques et heuristiques afin de résoudre ces systèmes. Ainsi, plusieurs organismes et en-

treprises ont adopté cette méthode, rendu populaire grâce à de puissants logiciels comme Gurobi (Gurobi Optimization, LLC, 2022). Dans un contexte académique, il s'agit d'une solution envisageable, puisque la plupart des outils permettant la résolution de problèmes à programmation entière offrent des licences académiques gratuitement, permettant ainsi d'utiliser la dernière technologie disponible la plus efficace.

Un autre avantage de cette technique de représentation de problème est de par sa flexibilité. Il est possible de facilement manipuler le système d'équations, ce qui est bénéfique pour plusieurs étapes de la conception. Par exemple, lors du développement, on peut facilement ajouter ou enlever des contraintes afin de valider des changements, ou bien tout simplement déboguer une contrainte spécifique. Dans la solution finale, si l'on connaît des particularités spécifiques au problème, ou bien des éléments de la réponse connus, on peut les ajouter pour accélérer et encadrer l'étape de résolution.

Dans le chapitre suivant (2.1), nous verrons comment effectuer la modélisation du problème de prédiction de structure d'ARN en utilisant la technique de la programmation entière.

1.3 Algorithmie appliquée à l'ARN

De nos jours, il existe des algorithmes efficaces permettant d'identifier les paires de bases canoniques et Wobble. Ces solutions sont principalement basées sur la théorie de l'énergie minimum libre (MFE), c'est-à-dire l'identification d'une structure où l'énergie résiduelle, après l'agencement des paires de bases, est à sa plus basse. Ces algorithmes se basent sur la programmation dynamique afin d'obtenir une matrice de probabilités d'agencement des paires de bases dans une complexité temporelle $O(n^3)$. Dans les principaux modèles utilisés à bon escient, on retrouve toujours le modèle de McKascill (McCaskill, 1990), ainsi que plus récemment le modèle de Turner (Turner et Mathews, 2010). L'efficacité de ceux-ci est grandement appréciée, et leur popularité est grandement due à la précision des paramètres énergiques et à leur excellente capacité de prédiction. Cela fait en sorte que plusieurs librairies sont basées sur ces modèles. On peut notamment trouver RNAstructure (Reuter et Mathews, 2010) ainsi que ViennaRNA (Lorenz *et al.*, 2011), qui dispose de plusieurs interfaces, notamment en Python.

Par contre, un autre aspect important de la prédiction de structure secondaire souvent oublié est l'omniprésence des interactions pseudo-nœuds. Dans plusieurs outils de prédiction, ces croisements de paires de bases sont omis, puisqu'elle présente un défi important en termes de complexité. Pourtant, ces interactions se retrouvent dans plusieurs structures d'ARN, et jouent un rôle majeur dans plusieurs de leurs fonctions, tel que le déphasage (*frameshift*) ribosomique et la régulation de l'épissage et de la traduction (van Batenburg *et al.*, 2001). Sous l'optique de l'énergie minimum libre (MFE), le problème de la modélisation de l'énergie se complexifie énormément, tout en diminuant sa précision. Lorsqu'on inclut tous les possibilités, le problème se présente comme étant d'une difficulté NP-Complet (Akutsu, 2000).

La plupart des solutions ce concentrent donc sous un sous-ensemble de famille de pseudo-nœuds. À titre de référence, pKnots, l'implémentation la plus générale de résolution de repliement avec pseudo-nœud, permet de détecter la présence de pseudo-nœuds dans une complexité temporelle de $O(n^6)$ et d'une complexité spatiale $O(n^4)$ (Rivas et Eddy, 1999). Dans un ensemble un peu plus restreint, CCJ permet la prédiction dans un temps $O(n^5)$ et un espace $O(n^4)$, en se concentrant sur les pseudo-nœuds de style *kissing hairpins*, Type H et de chaînes superposées (Chen *et al.*, 2009).

Du côté de la prédiction par maximisation de la précision espérée (MEA), ProbKnot a démontré qu'il était possible d'utiliser un algorithme qui maximise les probabilités des paires de bases canoniques insérées pour identifier la structure secondaire idéale avec pseudo-nœuds (Bellaousov et Mathews, 2010). Par la suite, la formulation de IPknot s'est avérée efficace pour prédire la structure secondaire contenant des pseudo-nœuds (Sato *et al.*, 2011). En utilisant la programmation entière, il est possible de prédire une structure secondaire avec pseudo-nœuds la plus probable, en profitant de puissants outils externes pour la résolution entière. Des travaux plus récents ont également utilisé une heuristique astucieuse pour résoudre de longues séquences en temps raisonnable, en combinant un seuil dynamique avec une approximation en temps linéaire de la partition de repliement de l'ARN (Sato et Kato, 2022).

Une autre approche à considérer est d'utiliser les données des structures que l'on connaît pour développer des modèles générés par de l'apprentissage machine. Cette tactique a été explorée par plusieurs groupes, notamment par l'équipe de MXfold2 (Sato *et al.*, 2021), en utilisant un modèle de thermodynamique comme score de repliement. Toutefois, on retrouve toujours plusieurs défauts dans ces modèles, et pour ce problème en particulier. Comme rapporté par certains experts, les données disponibles sur les structures d'ARNs sont peu nombreuses,

et hautement biaisées vers des familles spécifiques (Szikszai *et al.*, 2022). Par ailleurs, le recours à des données synthétiques ne suffit pas à combler cette faiblesse (Flamm *et al.*, 2022). Les propriétés locales des structures sont plutôt bien apprises, mais les propriétés globales, incluant le nombre de paires de bases total ainsi que la présence de pseudo-nœuds, sont rarement assimilées (Flamm *et al.*, 2022).

1.4 Travaux antérieurs (RNA-MoIP)

La base de cette recherche repose sur l'insertion de motifs structurels conservés dans l'objectif de raffiner la prédiction de la structure secondaire(Reinharz *et al.*, 2012). Ces travaux se constitue d'une implémentation Python 2, basée sur la programmation entière et résolue à l'aide du solveur propriétaire Gurobi(Gurobi Optimization, LLC, 2022). Le but principal était de sélectionner les motifs et les paires de bases canoniques qui maximisent la taille des motifs insérés, tout en minimisant les paires de bases retirées de la structure initiale. Cette optimisation devait se faire tout en respectant certaines contraintes pour assurer l'intégrité de la structure :

1. Aucune paire de bases isolées.
2. Aucun croisement entre les paires de bases.
3. Un minimum de 3 nucléotides non appariés à l'intérieur d'un regroupement de paires.
4. Un minimum de nucléotides appariés.

La base de données utilisée est basée sur la détection de réseaux similaires d'interactions non canoniques dans les structures 3D annotées d'ARN (Djelloul et Denise, 2008). L'ensemble des données est construit sur 888 structures 3D et contient 35 724 motifs de plusieurs catégories, comme des motifs d'épingles à cheveux, boucles intérieures et multiboucles. La possibilité d'insérer les motifs est basée sur une correspondance exacte de la séquence. Après avoir regroupé les séquences identiques, il en reste 4 696 uniques.

1.5 Objectifs du mémoire

Les travaux présentés dans cette recherche concernent l'évolution de ce programme à partir de cette base. Les principaux objectifs reposaient sur la mise à jour des motifs structuraux utilisés pour la prédiction, une approche itérative à la prédiction de structure secondaire ainsi qu'une possibilité de prédiction sans structure secondaire en entrée, en sélectionnant les paires de bases les plus probables selon le modèle d'énergie minimum libre.

En utilisant l'approche de la programmation entière, cela permet de réutiliser l'approche originale de RNA-MoIP. Pour proposer une solution plus concise, il est également possible de prendre en charge des structures secondaires avec pseudo-nœud par l'approche d'IPknot, combinant ainsi les forces de chacun des algorithmes. Il est d'ailleurs possible d'ajuster le poids de chacun des algorithmes. Pour faciliter la maintenabilité et assurer un support du programme, une réimplémentation du programme avec Python 3 a été de mise.

CHAPITRE II

MÉTHODOLOGIE

La définition du problème sera définie dans la Sec 2.1. Dans les sections 2.2 et 2.3, les deux parties intégrales de l'algorithme seront abordées. Par la suite, dans la section 2.4, les équations de programmation entière unifiées sont présentées. La prédiction précise des structures secondaires sur tous les ARNs non redondants connus avec une structure déterminée inférieure à 150 nucléotides est présentée dans la section 3.4, et comment les prédictions s'en sortent pour les interactions canoniques et non canoniques à l'intérieur des motifs dans les sections 3.5 et 3.6. Toutes les nouvelles additions au système d'équations seront identifiées de couleur verte.

2.1 Définition

Soit ω une séquence d'ARN, et Ω sa structure secondaire. Une paire de bases $(i, j) \in \Omega$ doit être canonique (G-C ou A-U) ou Wobble (G-U), et avoir $j - i > 3$. La classification Leontis-Westhof (LW) des paires de bases ARN (Leontis et Westhof, 2001) définit 12 géométries différentes possibles combinant deux côtés entre Watson-Crick (W), Hoogsteen (H), Sugar (S) et une orientation cis (c) ou trans (t). Les paires de bases canoniques et Wobble sont toutes de type cis Watson-Crick/Watson-Crick (cWW). En général, toute combinaison de nucléotides peut former n'importe quel type de paire de bases. La stabilité dans le modèle du plus proche voisin est obtenue à partir des paires de bases empilées (Turner et Mathews, 2010). Interdire les paires de bases solitaires implique formellement que si $(i, j) \in \Omega \Rightarrow (i - 1, j + 1) \in \Omega$ ou $(i + 1, j - 1) \in \Omega$.

La structure secondaire Ω peut être décomposée en un ensemble de structures sans pseudo-nœuds $\Omega^1, \ldots, \Omega^m$. Ω^q est sans pseudo-nœud s'il n'y a pas de croisement entre les paires de bases, formellement pour tous les $(i, j), (k, l) \in \Omega^q \Rightarrow i < k < l < j$ ou $k < i < j < l$. Plus concrètement, à titre d'exemple, une structure

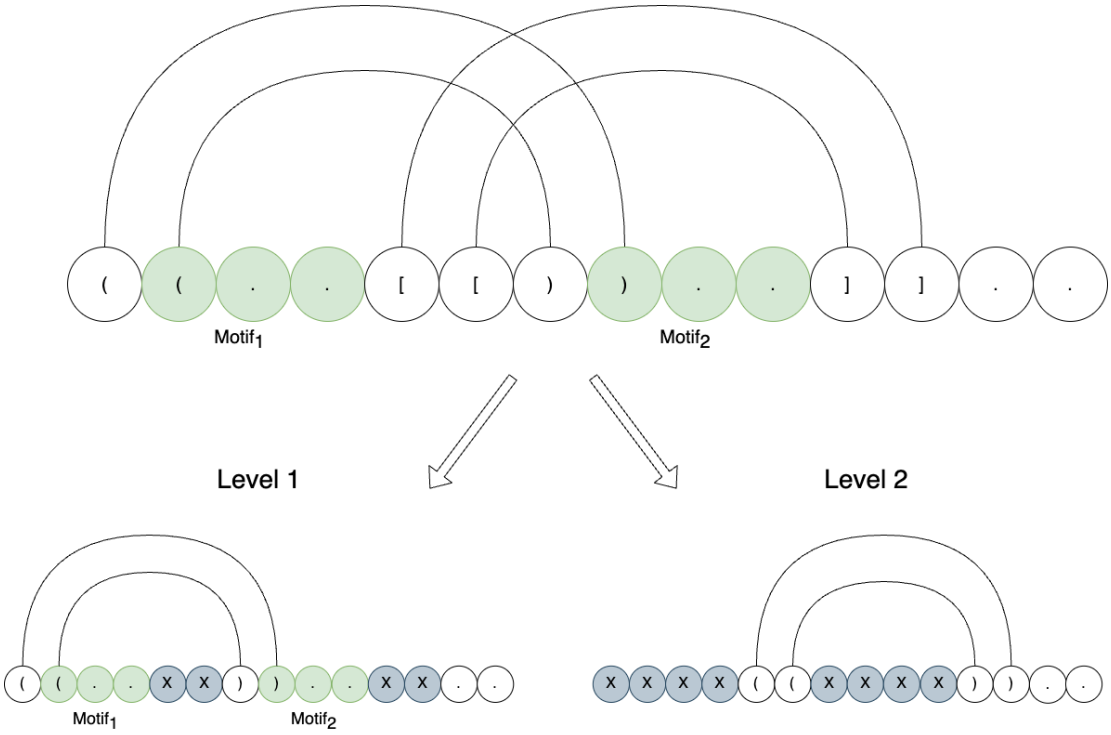


FIGURE 2.1 – **Exemple de décompositions de structure secondaire avec pseudo-nœud vers plusieurs structures sans pseudo-nœud**. À noter que l'insertion de motifs ne peut être faite que dans la structure du premier niveau, et que la présence de motif à une position donnée prévient la formation de paires de bases dans les niveaux supérieurs à cette position.

secondaire avec pseudo-nœud peut être décomposé de cette façon à la Fig. 2.1. Formellement, on peut représenter une décomposition de la façon suivante :

$$\texttt{((..[[))..]]..} \rightarrow \{\texttt{((..xx))..xx.} \in \Omega^1 + \texttt{xx..((xx..))..} \in \Omega^2\}$$

Le flux de travail complet de notre cadriciel est le suivant et schématisé dans la Fig. 2.2 :

1. Étant donné une séquence ω et une structure secondaire Ω, simultanément :
   (a) Décomposer Ω en structures sans pseudo-nœud, et pour chacune calculer une version contrainte d'un algorithme de repliement classique

pour obtenir une matrice de probabilité d'appariement de base (si aucune structure n'est fournie, l'algorithme s'exécute sans contraintes une fois).

(b) Additionner toutes les matrices ensemble.

(c) Trouver tous les emplacements possibles des motifs en utilisant la correspondance des motifs dans la séquence d'entrée ω. Les motifs ne peuvent s'insérer que dans le premier niveau de sous-structure.

2. Résoudre le modèle de programmation entière pour trouver la combinaison optimale de paires de bases avec des pseudo-nœuds et des motifs étant donné notre fonction objective détaillée dans la Sec. 2.4.3.

3. Itérer jusqu'à ce que :

(a) Deux itérations donnent la même solution.

(b) Un seuil de nombre d'itérations est atteint.

2.2 Motifs structuraux

La base de données fut construite à partir des données recueillies de Carnaval (Reinharz *et al.*, 2018). L'objectif de cette plateforme est de recueillir tous les ensembles de sous-graphes d'interactions canoniques et non canoniques communs aux différents organismes d'ARN. Pour être qualifiés, ces sous-graphes doivent avoir au moins 2 occurrences dans les ARNs, et soient le plus minimaux possible. En ciblant tous les sous-graphes des interactions locaux de tous les ARNs, il est possible d'obtenir 5278 séquences de motifs uniques. Ces séquences sont réparties à travers les 398 sous graphes (ou RINs, Recurrent Interaction Networks) de Carnaval. On retrouve la diversité des motifs de la base de données par rapport à leur nombre de brins dans la table 2.1.

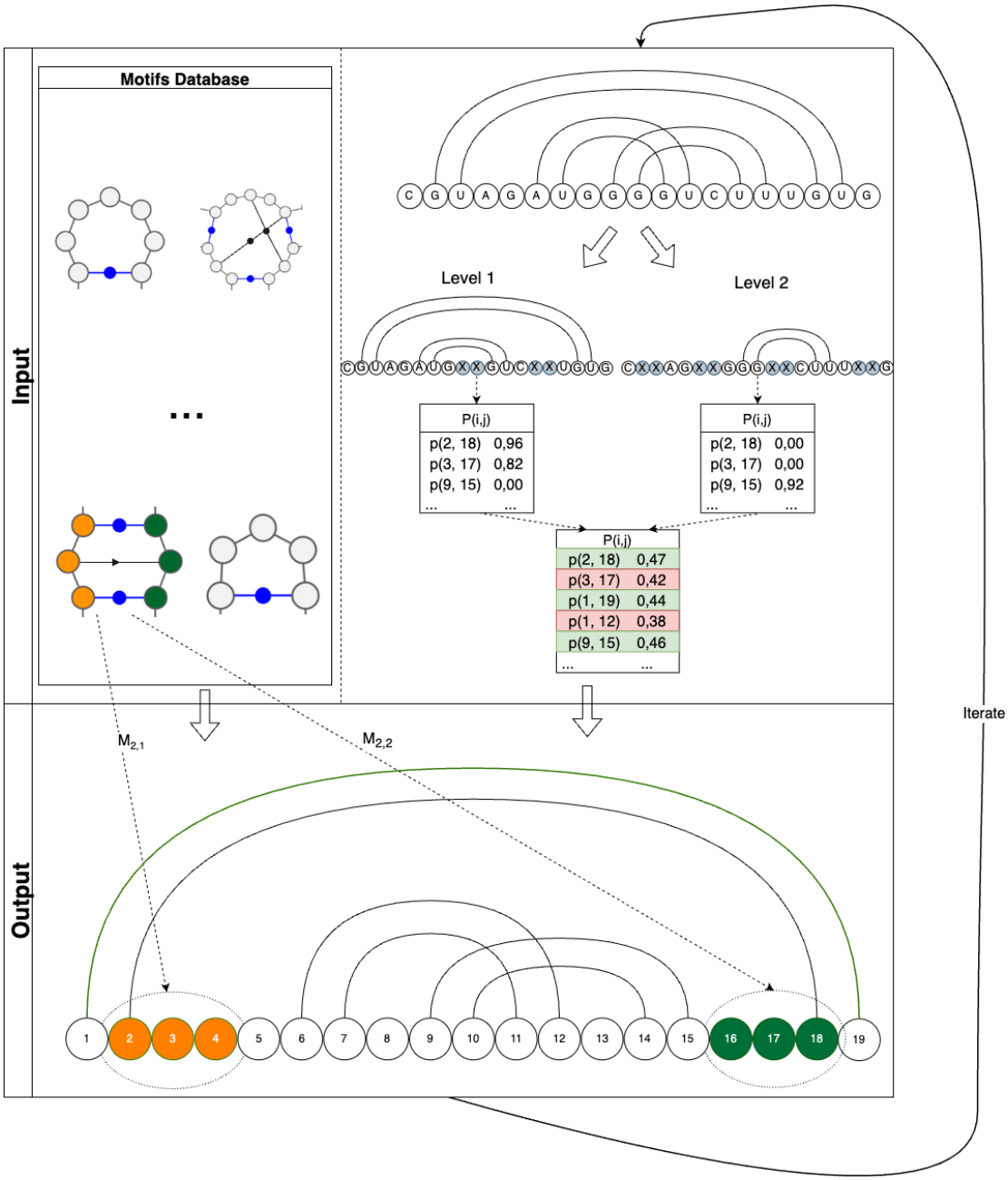
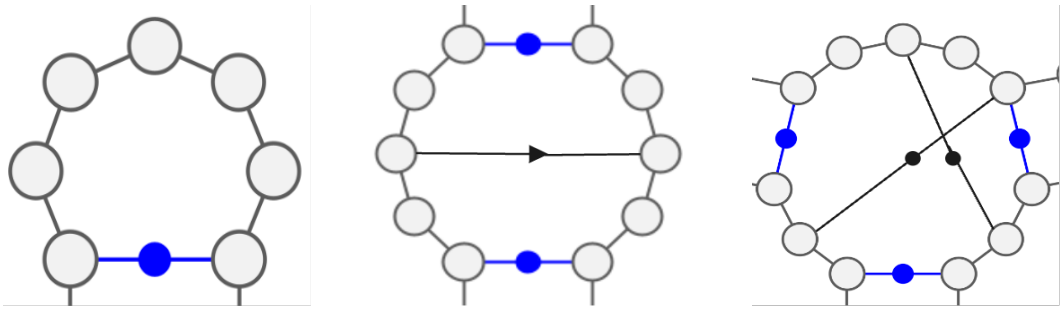


FIGURE 2.2 – **Flux de RNA-MoIP**. En haut à droite : la séquence avec une structure (qui peut être vide). La structure est décomposée en sous-structures sans pseudo-nœud, et pour chacune d'elles une matrice de PPB est calculée, puis additionnée ensemble. En haut à gauche : une base de données de motifs structuraux contenant des épingles à cheveux, des boucles et des renflements intérieurs, et des multijonctions. En bas : sortie d'une combinaison optimale entre une structure secondaire avec des pseudo-nœuds et des motifs insérés dans des emplacements compatibles avec la séquence. Chaque brin de motif doit empiler ou chevaucher par 1 position une paire de bases dans la structure secondaire.



(a) Tête à épingle (b) Boucle intérieur (c) Multiboucle

FIGURE 2.3 – **Exemples de motifs à 1, 2 et 3 brins**. Les interactions canoniques sont annotées en bleu et les interactions non canoniques en noir.

Nombres de brins	1	2	3	4 et +
Nombres de motifs	366	4038	543	331

TABLEAU 2.1 – **Diversité des motifs**. On considère que les motifs à 1 brin correspondent à des têtes à épingle, et des boucles intérieures et hernies pour des motifs à 2 brins. Les motifs à 3 brins et plus correspondent à des multiboucles.

2.3 Probabilité de paires de bases

Différents outils tels que ceux fournis par ViennaRNA (Lorenz *et al.*, 2011) et RNAstructure (Reuter et Mathews, 2010) peuvent précisément, dans un cadre thermodynamique, calculer les probabilités d'appariement des bases pour les structures sans pseudo-nœud. Afin de bien remédier au problème de la présence potentielle de pseudo-nœuds, suivant le modèle de IPknot (Sato *et al.*, 2011), la structure secondaire Ω est décomposée en un ensemble de structures sans pseudo-nœud $\Omega^1, \ldots, \Omega^m$. Pour chacune des sous-structures, une matrice de probabilité de paires de bases (PPB) peut être calculée de telle sorte que les paires de bases dans cette sous-structure soient appliquées comme des contraintes dures, et que toute position dans une paire d'une autre sous-structure est interdite. Dans ce contexte, puisque l'on obtient des sous-structures sans pseudo-nœud, le calcul des PPBs

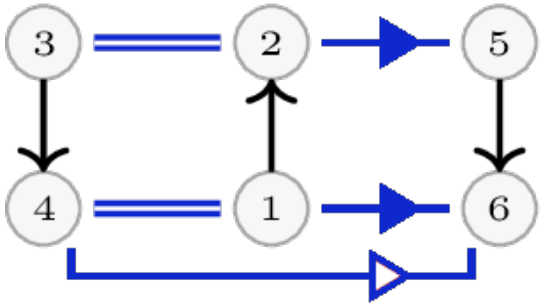


FIGURE 2.4 – **Exemple de RIN présent dans la base de données de Carnaval**.

demeure dans une complexité temporelle $O(n^3)$. Dans la formulation de la programmation entière, le poids de la paire de bases (i, j) sera la pseudo-probabilité $p(i, j) = p^1_{i,j} + \cdots + p^m_{i,j}$. Cette étape est démontrée en haut à droite de la Fig 2.2. Notez que si lors de l'évaluation des PPBs les paires sont considérées comme des contraintes dures, elles doivent être préservées. Cependant, il n'y a pas de telle condition dans le modèle entier.

2.4 Modèle de programmation entière

Le modèle de programmation en nombres entiers est assez complexe et je reproduis ici toutes les équations par souci d'exhaustivité. La Sec. 2.4.1 décrit comment les motifs sont encodés. La Sec. 2.4.2 liste toutes les variables du modèle. Puis la fonction objective est détaillée dans la Sec. 2.4.3. Les contraintes concernant le placement des paires de bases dans les sections 2.4.4 et 2.4.5, et celles concernant l'insertion des motifs sont dans les sections 2.4.6 et 2.4.7. Enfin, les deux modèles coexistent grâce à la dernière contrainte dans la section 2.4.8.

2.4.1 Paramètres

On désigne une séquence d'ARN par ω et par Ω une structure secondaire compatible avec celle-ci. ω_i est le nucléotide en position i et doit être dans $\{A, C, G, U\}$. La structure peut être vide ou peut contenir des interactions de croisement. J'utilise $n = ||\omega||$ comme longueur de la séquence.

Chaque motif de la base de données peut être composé d'un ensemble de brins de séquence différents (par exemple, une boucle intérieure à deux brins). Les équations feront la différence entre les épingles à cheveux (1 brin), les boucles intérieures et les renflements (2 brins), et les multijonctions (3 brins ou plus). Mot^j est l'ensemble des motifs avec j-brins et chacun est composé de sa liste de brins. Pour tout motif $x \in Mot$, la longueur $|x|$ représente le nombre de nucléotides qu'il contient. Formellement, une position x_a^b dans un brin peut être A, C, G, U ou le joker *, et on a :

$$Mot^j = \{x \mid x := [(x_1^1, \cdots, x_{k_1}^1), \cdots, (x_1^j, \cdots, x_{k_j}^j)] \text{ et } \exists \text{ un } match \text{ de } x \text{ dans } \omega\}.$$

Les brins sont ordonnés dans l'ordre de $5'$ à $3'$ de la séquence dont ils sont extraits. Le modèle doit savoir où le i-ième brin de tout motif composé de j parties peut être inséré. Ceux-ci peuvent également être de longueurs différentes. L'ensemble Seq_i^j sera un ensemble de triplets avec le nom du motif et les première et dernière positions où le i-ième brin de ce motif peut être inséré. Le même brin peut être placé à plusieurs endroits différents. Formellement :

$$Seq_i^j = \{(x, a, a + k_i - 1) \mid x \in Mot^j \text{ and } x_1^i, \ldots, x_{k_i}^i = \omega_a, \ldots, \omega_{a+k_i-i}\}.$$

2.4.2 Variables

Pour rester dans la lignée de l'implémentation précédente de RNA-MoIP, deux types de variables binaires sont crées. Premièrement, pour les motifs, 4 paramètres doivent être suivis : (1) x le nom du motif (2) s'il s'agit du i-ième brin de ce motif, et (3,4) l'intervalle k, l où il est inséré. Ceci sera conservé par la variable $C_{k,l}^{x,i}$ représentant l'insertion du i-ième élément du motif x à la position (k, l) de la séquence. Une telle variable existe pour chaque élément de chaque ensemble Seq_i^j. Deuxièmement, pour chaque paire de positions où la PPB est supérieure à un certain seuil (Fixé à 0.05 dans nos expériences), le modèle doit savoir s'il est instancié et à quel niveau. À partir de la décomposition de la structure (voir Fig 2.2), on associe dorénavant les paires à leur niveau de structure. Le premier niveau correspond à la sous-structure sans croisement pseudo-nœud. Puis, chaque niveau supérieur contiendra lui-même une sous-structure sans pseudo-nœud, et chaque paire de ce niveau devra croiser au moins une paire de bases dans le niveau inférieur. La variable binaire $D_{u,v}^q$ vaudra 1 s'il existe une paire de bases entre ω_u et ω_v au niveau q. L'ensemble $\mathcal{B}$ contiendra toutes les paires de positions (u, v) avec une paire de bases potentielle.

2.4.3 Objectif

Intuitivement, on veut maximiser les probabilités des paires de pseudo-bases comme la quantité d'information dans la séquence. Provenant de la première version du cadriciel (Reinharz *et al.*, 2012), ceci est réalisé en maximisant le carré de la longueur des motifs insérés, ce qui pousse à insérer la plus petite quantité de motifs les plus grands possibles. Comme dans la Sec. 2.4.1, le nombre de nucléotides dans le motif x est noté $|x|$. Pour chaque paire de bases potentielle entre les positions (u, v), la probabilité pseudo-PPB $p(u, v)$ (voir Sec. 2.3) est utilisée. Un facteur

de 10 a été ajouté empiriquement. Un paramètre α sera utilisé pour combiner la maximisation des pseudo-PPBs et l'insertion de motifs. Un poids β^q est utilisé pour équilibrer les différents niveaux de pseudo-nœuds. En suivant IPknot, les poids sont fixés à $\beta^1 = 0.5, \beta^2 = 0.25, \beta^3 = 0.125$ et $\beta^4 = 0.0625$. Formellement, la fonction objective est la suivante :

$$\max \alpha \sum_{x \in Mot^j} \left(\underbrace{(|x|)^{\,2}}_{\text{Longueur du motif}} \times \sum_{(x,k,l) \in Seq_1^j} \underbrace{C_{k,l}^{x,1}}_{\text{Motif x inséré à la position (k,l)}} \right) + 10(1-\alpha) \times \sum_{(u,v) \in \mathcal{B}} \sum_{q=1}^{m} \underbrace{D_{u,v}^q}_{\text{Paire de bases (u,v)}} \; \underbrace{p(u,v)}_{\text{Probabilité de la paire de bases (u,v)}} \; \underbrace{\beta^q}_{\text{Poids du niveau q}} \tag{2.1}$$

2.4.4 Contraintes des paires de bases

Trois équations assurant une structure adéquate sont originales de la première version de RNA-MoIP, avec une légère adaptation, identifiée en vert. La première équation garantie que chaque position ne se trouve que dans une seule paire de bases (Eq. 2.2). Les deux propriétés supplémentaires suivantes sont appliquées : les paires de bases doivent être empilées sur le même niveau (Eq. 2.3 et 2.4) et une proportion d'au moins θ des positions doivent être dans une paire de bases (Eq. 2.5). Notez que naturellement, en raison des pondérations dans la fonction objective, la plupart des paires de bases se concentreront sur les niveaux inférieurs.

$$\forall 1 < u < n : \sum_{q}^{m} \sum_{\substack{(\tilde{u},\tilde{v}) \in B \\ \tilde{u}=u \vee \tilde{v}=u}} D_{\tilde{u},\tilde{v}}^q \leq 1 \tag{2.2}$$

$$\forall 1 < q \leq m,\ \forall 1 < i < n :$$

$$\sum_{\substack{(u,v)\in B \\ u=i-1 \vee u=i+1}} 1 - \sum_{\substack{(u,v)\in B \\ u=i-1 \vee u=i+1}} D^q_{u,v} \geq \sum_{\substack{(u,v)\in B \\ u=i}} 1 - \sum_{\substack{(u,v)\in B \\ u=i}} D^q_{u,v} \tag{2.3}$$

$$\forall 1 < q \leq m,\ \forall 1 < i < n :$$

$$\sum_{\substack{(u,v)\in B \\ v=i-1 \vee v=i+1}} 1 - \sum_{\substack{(u,v)\in B \\ v=i-1 \vee v=i+1}} D^q_{u,v} \geq \sum_{\substack{(u,v)\in B \\ v=i}} 1 - \sum_{\substack{(u,v)\in B \\ v=i}} D^q_{u,v} \tag{2.4}$$

$$2\sum_{q}^{m} \sum_{(u,v)\in B} D^q_{u,v} \geq \omega n \tag{2.5}$$

2.4.5 Contraintes de la décomposition

Nouvellement à la présente version, deux équations sont nécessaires afin de respecter la décomposition des sous-structures. Chaque niveau q contient une structure sans pseudo-nœud (Eq. 2.6), et les paires de bases sont ajoutées à un niveau si et seulement si elles croisent une autre paire de bases dans chaque niveau inférieur (Eq. 2.7).

$$\forall 1 < q \leq m,\ \forall 1 \leq i < j < k < l \leq n : D^q_{i,j} + D^q_{k,l} \leq 1 \tag{2.6}$$

$$\forall 1 < q \leq m,\ \forall (u,v) \in B : \sum_{i<u<j<v}^{B} D^{q-1}_{i,j} + \sum_{u<\tilde{i}<v<\tilde{j}}^{B} D^{q-1}_{\tilde{i},\tilde{j}} \geq D^q_{\tilde{u},\tilde{v}} \tag{2.7}$$

2.4.6 Contraintes sur l'insertion des motifs

En suivant la formulation originale de RNA-MoIP, je reproduis ici toutes les équations nécessaires à l'insertion des motifs. Bien que la plupart d'entre elles soient similaires, il existe une différence notable. L'une des principales conditions pour l'insertion d'un motif est qu'il doit être empilé ou chevauche la dernière paire de bases d'un brin. Comme la base de données de motifs est définie sur les boucles d'une structure secondaire de pseudo-nœud, seules les paires de bases du premier niveau sont considérées. Je donne un aperçu de chaque équation, mais plus de détails peuvent être trouvés dans la publication originale (Reinharz *et al.*, 2012).

L'insertion des épingles à cheveux qui ne sont composées que d'un seul brin est contrainte par l'équation 2.8, qui a été modifiée pour ne considérer que les paires de bases du niveau 1. L'insertion de boucles intérieures et de renflements doit d'abord garantir que les brins sont placés dans des positions acceptables (Eq. 2.9) et que le motif doit remplir au moins 2 positions non appariées, garantissant ainsi l'ajout d'informations au système (Eq. 2.10).

$$\forall_{(x,k,l)\in Seq_1^1} : \; C_{k,l}^{x,1} \leq \sum_{\substack{(u,v)\in B \\ k-1\leq u\leq k \wedge l\leq v\leq l+1}} (D_{u,v}^1) + \sum_{\substack{(\tilde{x},\tilde{k},\tilde{l})\in Seq_1^2 \\ \tilde{l}=k-1}} C_{\tilde{k},\tilde{l}}^{\tilde{x},1} + \sum_{\substack{(\tilde{x},\tilde{k},\tilde{l})\in Seq_2^2 \\ \tilde{k}=l+1}} C_{\tilde{k},\tilde{l}}^{\tilde{x},2} \quad (2.8)$$

$$\forall (u,v) \in B, \; \forall x \in Mot^2 : \; -n(1-D_{u,v}^1) \leq \sum_{\substack{(x,k,l)\in Seq_1^2 \\ l<u\vee v<k}} C_{k,l}^{x,1} - \sum_{\substack{(x,k,l)\in Seq_2^2 \\ l<u\vee v<k}} C_{k,l}^{x,2} \leq n(1-D_{u,v}^1) \quad (2.9)$$

$$\forall (x,k,l) \in Seq_1^2, \forall (x,\tilde{k},\tilde{l}) \mid \left[\tilde{k} > l \wedge 2 \cdot \sum_{\substack{(u,v)\in B \\ k \le u \le l \wedge \tilde{k} \le v \le \tilde{l}}} 1 + \sum_{\substack{(u,v)\in B \\ k \le u \le l \oplus \tilde{k} \le v \le \tilde{l}}} 1 \ge l - k + \tilde{l} - \tilde{k} + 1 \right] \in Seq_2^2 : C_{k,l}^{x,1} + C_{\tilde{k},\tilde{l}}^{x,2} \le 1 \tag{2.10}$$

L'admissibilité de l'insertion dans les jonctions à k voies est décidée dans l'Eq. 2.11, en s'assurant que chaque brin peut être atteint sans traverser les paires de bases du premier niveau. Ceci est équivalent à l'Eq. 2.9 pour les boucles intérieures.

$$\forall j \ge 3, \forall (u,v) \in B : \; -n(1 - D_{u,v}^1) \le (j-1) \cdot \sum_{\substack{(x,k,l)\in Seq_1^j \\ u \le k \le l \le v}} C_{k,l}^{x,1} - \sum_{\substack{1<i\le j \\ (x,k,l)\in Seq_i^j \\ u \le k \le l \le v}} C_{k,l}^{x,i} \le n(1 - D_{u,v}^1) \tag{2.11}$$

2.4.7 Contraintes sur l'ordre des motifs

Une caractéristique importante de la structure des ARN est que leur séquence est ordonnée, de l'extrémité $5'$ à l'extrémité $3'$, et qu'elle n'est pas symétrique. Dans un motif, un ordre est défini sur les brins qui suivent cette direction. Le modèle contraint l'endroit où un brin dans un motif peut être placé étant donné l'insertion du brin précédent (Eq. 2.12) ou du brin suivant (Eq. 2.13) du même motif. Une considération importante est qu'à la fin, il doit exister une décomposition mutuellement exclusive des brins tels que chaque motif inséré soit complet, même si de nombreuses copies sont trouvées (Eq. 2.14).

$$\forall\ 1 < i \leq j,\ \forall (x,k,l) \in Seq_i^j :\ C_{k,l}^{x,i} \leq \sum_{\substack{(x,\tilde{k},\tilde{l}) \in Seq_{i-1}^j \\ \tilde{l} < k-5}} C_{\tilde{k},\tilde{l}}^{x,i-1} \tag{2.12}$$

$$\forall\ 1 \leq i < j,\ \forall (x,k,l) \in Seq_i^j :\ C_{k,l}^{x,i} \leq \sum_{\substack{(x,\tilde{k},\tilde{l}) \in Seq_{i+1}^j \\ l+5 < \tilde{k}}} C_{\tilde{k},\tilde{l}}^{x,i+1} \tag{2.13}$$

$$\forall\ j > 1,\ \forall x \in Mot^j,\ \forall 1 < i \leq j : \sum_{(x,k,l) \in Seq_1^j} C_{k,l}^{x,1} - \sum_{(x,\tilde{k},\tilde{l}) \in Seq_i^j} C_{\tilde{k},\tilde{l}}^{x,i} = 0 \tag{2.14}$$

2.4.8 Contrainte sur la combinaison des approches

Enfin, pour unifier les deux modèles, il est important d'éviter les conflits entre les motifs et les paires de bases aux différents niveaux. Puisque les motifs représentent des cycles locaux, on prévient également la formation de paire de longue distance à l'intérieur des motifs. C'est le rôle de l'Eq. 2.15.

$$\forall\ 1 < i \leq j,\ \forall (x,k,l) \in Seq_i^j :\ \sum_{q=1}^{m} \sum_{\substack{(u,v) \in B \\ k \leq u \leq l \vee \\ k \leq v \leq l}} D_{u,v}^q \leq \left(1 - C_{k,l}^{x,i}\right) \cdot \sum_{q=1}^{m} \sum_{\substack{(u,v) \in B \\ k \leq u \leq l \vee \\ k \leq v \leq l}} 1 \tag{2.15}$$

CHAPITRE III

RÉSULTATS

3.1 Implémentation

Le cadre de programmation en nombres entiers est implémenté en Python 3 sous forme de librairie. Celle-ci permet l'utilisation de différents outils pour la résolution du modèle de programmation entière, dépendamment des licences disponibles à l'utilisateur. Dans cette étude, j'ai utilisé le solveur open source OR-Tools (OR-Tools, 2021). Des instructions pour utiliser le solveur propriétaire Gurobi (Gurobi Optimization, LLC, 2022) ainsi que la librairie MIP (ICEB, 2021), qui interface entre autres le solveur open source CBC (COIN-OR, 2022), sont également fournies. J'ai effectué nos séries de tests sous Ubuntu 21.04 sur un processeur Intel Xeon W-2295 avec 512GB 8x64GB DDR4 2933 MHz. Le code source, les données et les résultats sont disponibles sur `https://gitlab.info.uqam.ca/cbe/RNAMoIP` sous une licence MIT.

Afin de rendre le programme plus accessible, un serveur web a été développé et rendu disponible à l'adresse `https://rnamoip.cbe.uqam.ca`. L'utilisateur peut voir les résultats de prédiction disponibles à titre d'exemple, ou faire sa propre prédiction, en fournissant une séquence et optionnellement une structure secondaire. Il peut également ajuster certains paramètres de RNA-MoIP, comme la valeur α ou le niveau maximum de croisement de paires. Une fois la prédiction asynchrone terminée, un tableau de bord présente diverses informations relatives à la structure prédite trouvée. Nous pouvons trouver une représentation 2D de la structure avec les différents motifs insérés, construits à l'aide de Varna (Darty *et al.*, 2009). L'utilisateur peut également basculer entre plusieurs solutions trouvées par le solveur si présentes et peut voir plus en détail toutes les occurrences qui correspondent à chaque motif trouvé dans leur onglet respectif, avec toutes leurs interactions canoniques et non canoniques. Il est même possible de visualiser et de comparer la structure 3D des occurrences après sélection.

3.2 Dataset

Pour l'étalonnage, toutes les structures d'ARN entre 20 et 150 nucléotides dans la banque des PDBs (Berman *et al.*, 2000) ont été sélectionnées, en filtrant les séquences identiques. Pour éviter les redondances moléculaires, j'ai conservé une structure par classe non redondante telle que définie par le BGSU RNA Structure Atlas (BGSU RNA, 2022), version 3.208.

Les paires de bases canoniques de la structure secondaire peuvent être déconvoluées de différentes manières en une structure principale sans nœud et un ensemble de pseudo-nœuds de complexité croissante (Smit *et al.*, 2008). Cette organisation de la structure secondaire a été déterminée en utilisant RNApdbee (Antczak *et al.*, 2014; Zok *et al.*, 2018). Le jeu de données de référence est composé des 101 structures restantes ayant au moins un pseudo-nœud.

À partir de ce jeu de données, j'ai analysé la proportion de nucléotides appariés à une paire de bases dans les structures réelles, qui est présentée dans la Fig. 3.1. Ainsi, j'ai choisi de fixer le paramètre θ dans l'équation 2.5 à 25%, afin de couvrir 100 des 101 PDBs avec pseudo-nœuds.

3.3 Résolution

La version 2.5.0a5 de ViennaRNA est utilisée pour calculer les matrices de probabilité des paires de bases. Les conditions de terminaison sont fixées à un maximum de 3 itérations, ou deux itérations avec les mêmes résultats. Un temps de $10^4 s$ a été alloué pour chaque prédiction et séquence. Dans notre cas d'utilisation avec le solveur OR-Tools, la meilleure solution identifiée dans le temps alloué est retournée lorsque la limite de temps est atteinte. Bien que plusieurs solutions optimales puissent exister pour une formulation entière, la première obtenue a été utilisée.

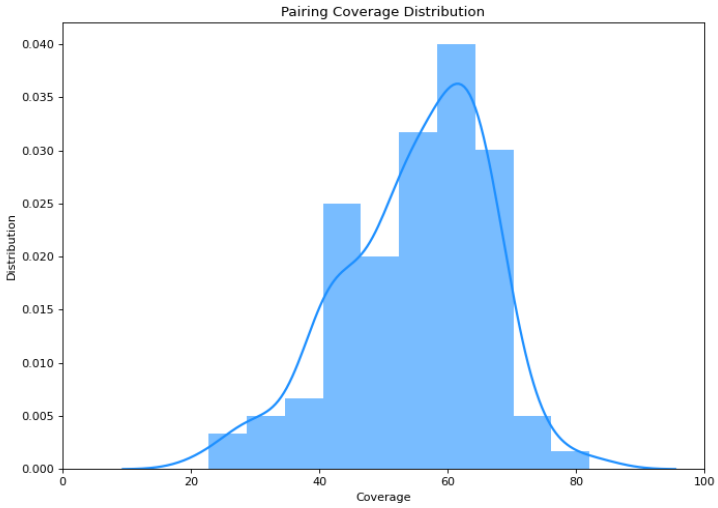


FIGURE 3.1 – **Proportion des nucléotides appariés**. Dans les 101 structures en entrées, 1 seul (4V6W-A8) est en dessous du seuil fixé à 25%.

3.4 Insertion de motifs

Pour évaluer la capacité de RNA-MoIP à prédire la structure secondaire, l'ensemble des vrais positifs (VP) est constitué de paires de bases canoniques et de Wobble correctement prédites. L'ensemble des faux positifs (FP) correspond aux paires de bases canoniques prédites qui ne sont pas dans la structure secondaire du PDB, et l'ensemble des faux négatifs (FN) correspond aux paires de bases canoniques du PDB manquantes dans la prédiction. La valeur de précision (VPP : $VP/(VP+FP)$, la sensibilité (STY : $VP/(VP+FN)$) et la mesure F ($2VPP \cdot STY/(VPP+STY)$) sont utilisées comme métriques et présentées dans la Fig. 3.2. Je compare les résultats de RNAfold et RNA-MoIP avec différentes valeurs de α. Lorsque $\alpha = 0$, aucun motif n'est considéré, et le modèle est équivalent à l'implémentation d'IPknot. Notez que la sensibilité de RNAfold ne peut pas atteindre 1, puisque seules des structures de pseudo-nœuds sont dans l'ensemble de référence, et que RNAfold ne peut prédire ces interactions croisées. Et c'est ce

que l'on peut observer dans les résultats, où on trouve une sensibilité inférieure à presque toutes les valeurs de α. D'autre part, les prédictions de RNAfold sont en général plus sensibles que le modèle entier, surtout lorsqu'aucun motif n'est utilisé. Les valeurs moyennes sur tous les modèles finis sont présentées dans le tableau 3.1.

On remarque également que notre implémentation d'IPknot ($\alpha = 0$) obtient la meilleure sensibilité globalement. Puisque le seul objectif est de maximiser la somme des probabilités des paires de bases insérées, beaucoup de paires de bases se retrouvent dans la prédiction. Cela se reflète alors dans la métrique de précision, où l'on constate que plusieurs paires de bases insérées ne se retrouvent pas dans la structure finale. En combinant avec l'objectif d'insertions de motifs, la sensibilité reste relativement haute, et la précision s'en trouve grandement améliorer. La mesure F optimal, équilibrant la quantité de paires de bases prédites et leur sensibilité, est atteinte avec $\alpha = 0.15$, en complétant le modèle entier avec des informations sur les motifs.

	RNAfold	$\alpha = 0$	$\alpha = 0.05$	$\alpha = 0.1$	$\alpha = 0.15$	$\alpha = 1$
VPP	0.63	0.57	0.62	0.654	**0.671**	0.667
STY	0.598	**0.683**	0.666	0.632	0.625	0.533
F1	0.607	0.616	0.638	0.638	**0.641**	0.588

TABLEAU 3.1 – **Sommaire des résultats des prédictions**.

Les statistiques présentées précédemment sont calculées sur l'ensemble de la structure secondaire. Nos modèles autorisent jusqu'à 2 niveaux de croisement entre les paires de bases alors que 97 des structures étudiées ne présentent qu'un seul niveau d'interactions croisées. Aucune surprédiction du niveau de pseudo-nœuds n'a été observée, comme le montre le tableau 3.2. En fait, les pseudo-nœuds sont généralement sous-prédit. Lorsqu'aucun motif n'est inséré, environ 25% des structures

ont un niveau de pseudo-nœuds trop basse, et jusqu'à 50% lorsque des motifs sont ajoutés au modèle. Néanmoins, l'amélioration de la VPP et de plus de 3% de la mesure F indique que, même si aucun pseudo-nœud n'est prédit, la structure est beaucoup plus précise.

α	0	0.05	0.1	0.15	1
Niveau de structure maximal trop bas	25	51	62	68	86
Niveau de structure maximal correct	76	50	39	33	15

TABLEAU 3.2 – **Prédiction du niveau maximal de pseudo-nœuds**. Lorsque α est augmenté, on sous-estime le nombre de pseudo-nœuds présentes dans la structure. La complexité du modèle entier augmente et rend plus difficile la recherche d'une solution optimale réalisable à temps. Aucune surprédiction du niveau maximal n'a été observée.

3.5 Interactions canoniques et de Wobble

Pour chaque motif inséré, j'ai récupéré dans l'atlas de la structure de l'ARN toutes les interactions canoniques et de Wobble aux positions insérées pour construire nos exemples positifs. Notez que ces interactions peuvent être croisées à l'intérieur des boucles et n'ont pas besoin d'être empilées, elles ne font donc pas nécessairement partie de la structure secondaire. Puisqu'une séquence de motifs dans notre base de données peut correspondre à différentes sous-structures, celle qui présente la meilleure correspondance structurelle a été utilisée dans cette section et les suivantes. Une métrique sur le ratio des paires de bases de la structure réelle correctement prédites est calculée et moyennée sur tous les motifs dans toutes les structures et présentée dans la Fig. 3.4. À l'intérieur des motifs, un ratio de plus de 40% est atteinte, ce qui correspond à un taux légèrement inférieur pour la prédiction des paires canoniques et de Wobble dans les motifs que dans la structure secondaire de pseudo-nœud.

α	0	0.05	0.1	0.15	1
Nombre de Motifs	0	826	912	929	904
Pas d'interactions canoniques	0	382	426	456	483
Pas d'interactions non canoniques	0	724	811	829	811

TABLEAU 3.3 – **Nombre de motifs insérés dans tous les PDBs évalués**. Inclus le nombre d'occurrences où aucune interaction (canonique ou non canonique) n'a été à l'emplacement répertorié du motif dans le PDB.

3.6 Interactions non canoniques

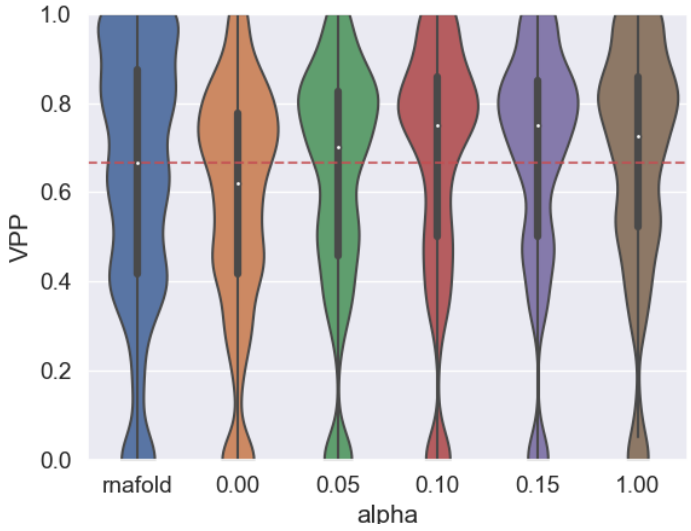
Bien qu'elles soient minoritaires et difficiles à prévoir, il a été démontré que les interactions non canoniques sont nécessaires à de nombreuses fonctions de l'ARN. Comme dans la section précédente, pour chaque motif, l'ensemble positif consiste en l'ensemble de toutes les interactions non canonique entre les positions insérées. La distribution de VPP, STY et F1 sont présentées dans la Fig. 3.5, ce qui montre à quel point les motifs sont encore limités pour prédire ces informations à granularité fine. Bien que cela semble dramatique, cela peut s'expliquer de deux manières différentes. Premièrement, leur nombre est vraiment faible, comme le montre la Fig. 3.6. L'axe Y montre qu'à ces positions, rarement plus de 4 interactions non canoniques sont présentes par ARN, et rarement plus de 2 dans les motifs insérés. Deuxièmement, on sous-estime les interactions non canoniques puisque beaucoup ne peuvent pas être prédites par notre modèle. En plus de celles qui se trouvent à des endroits où aucun motif n'est prédit, beaucoup peuvent relier des motifs entre eux, mais elles ne peuvent pas être trouvées dans l'ensemble de données utilisé. On peut voir dans la table 3.3 les interactions des PDBs représentées à l'intérieur des motifs insérés.

α	0	0.05	0.1	0.15	1
Nombre de PDBs	0	0	2	3	13

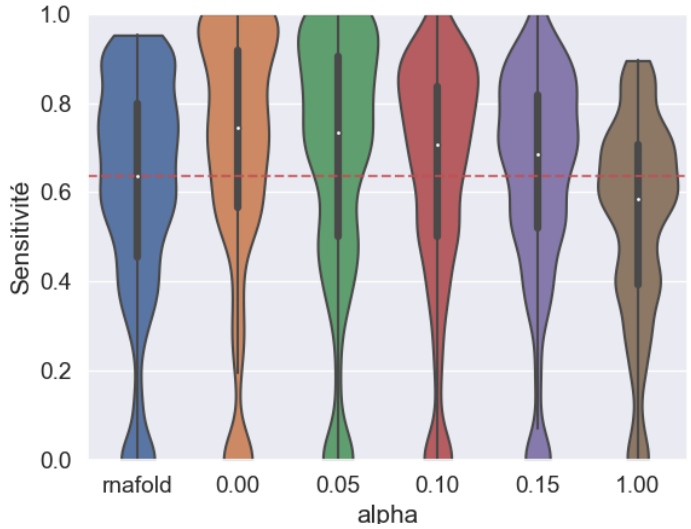
TABLEAU 3.4 – **Nombre de PDBs n'ayant pas obtenu un résultat optimal dans le temps alloué**. Une solution jugée satisfaisante par le solveur est alors retournée dans ce cas.

3.7 Performance

La programmation en nombres entiers est connue pour être NP-Complet, mais des décennies d'optimisation ont permis de tirer parti d'implémentations efficaces pour exprimer des modèles complexes. Il faut rappeler qu'un maximum de temps de 10^4 secondes a été appliqué. Lorsque le temps maximal est atteint, le solveur utilisé retourne la meilleure solution trouvée dans le temps répartie, sans toutefois garantir qu'il s'agit de la solution maximale du problème. La figure 3.7 démontre le temps d'exécution en secondes pour chaque PDB, en fonction de leur nombre de nucléotides. La table 3.4 montre la quantité de solutions ayant atteint la limite de temps imposée. Avec $alpha = 0.1$, les deux PDBs ayant dépassé la limite, 3G9C-Q et 4PR6-B, possèdent respectivement 141 et 144 nucléotides. D'autres heuristiques comme celles développées par IPknot pour les longues séquences pourraient être utilisées, au prix d'une diminution de la précision. Je pense que des gains optimaux seraient obtenus en optimisant l'emplacement où l'on permet l'insertion des motifs.



(a) VPP



(b) STY

FIGURE 3.2 – **Prédiction de la structure secondaire avec pseudo-nœuds**. Comparaison des résultats pour RNAfold (ne peut pas prédire les interactions croisées), sans insertion de motifs (IPknot, ou $\alpha = 0$), et pour différentes valeurs de α. Lorsque $\alpha > 0$, toutes les paires de bases qui se retrouvent dans les motifs sont comptées comme de vrais positifs.

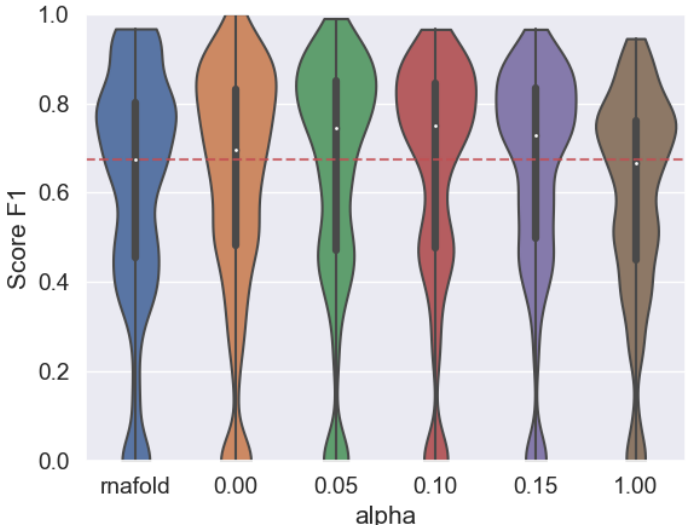


FIGURE 3.3 – **Prédiction de la structure secondaire avec pseudo-nœuds**. Comparaison des résultats pour le score F1.

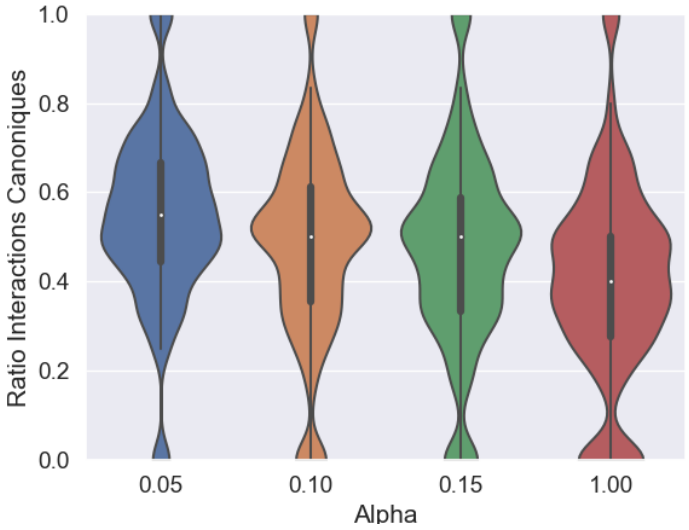


FIGURE 3.4 – **Ratio des paires de bases canoniques et de Wobble dans les motifs**. Pour des valeurs de α de 0.05, 0.1, 0.15, plus de 50% des paires de bases canoniques et Wobble dans les motifs sont généralement capturées.

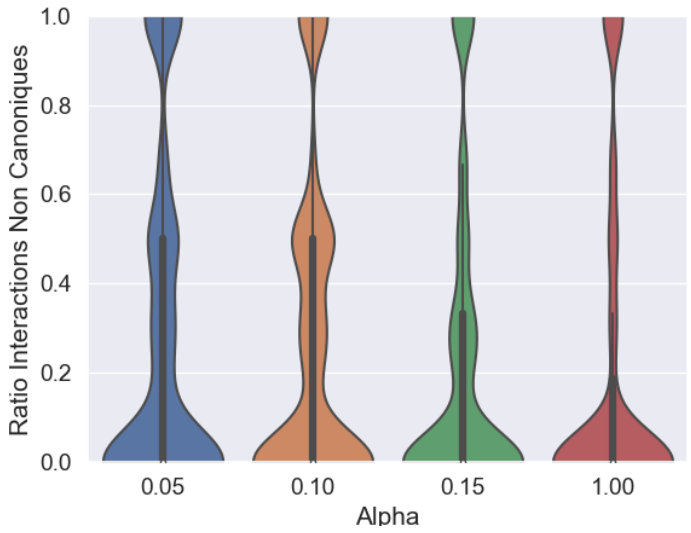


FIGURE 3.5 – **Ratio des paires non canoniques correctement prédites dans les motifs**. Les vrais positifs sont les paires de bases non canoniques aux positions où un motif est inséré dans la séquence. Elles composent au maximum 15% des interactions dans les motifs insérés, et sont difficiles à prédire.

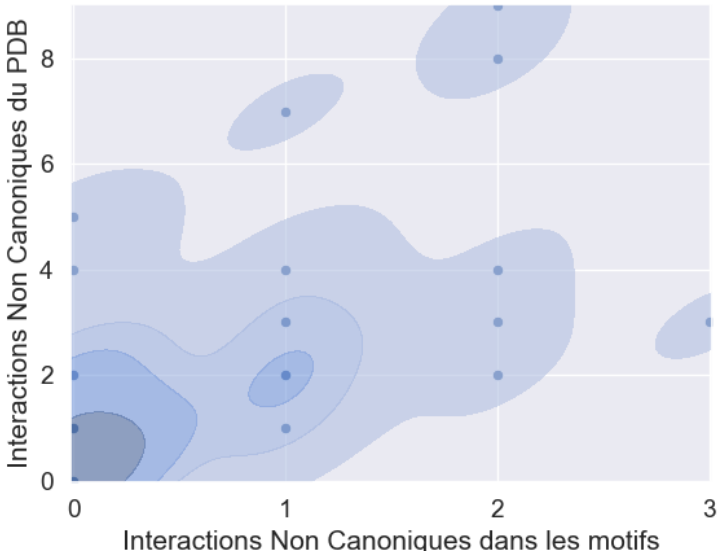


FIGURE 3.6 – **Interactions non canoniques**. Distribution du nombre d'interactions non canoniques qui sont observées aux emplacements des motifs insérés, à $\alpha = 0.1$. Sur l'axe des y, le nombre d'interactions dans la structure réelle aux positons des motifs, sur l'axe des x, combien sont annotées dans le motif inséré.

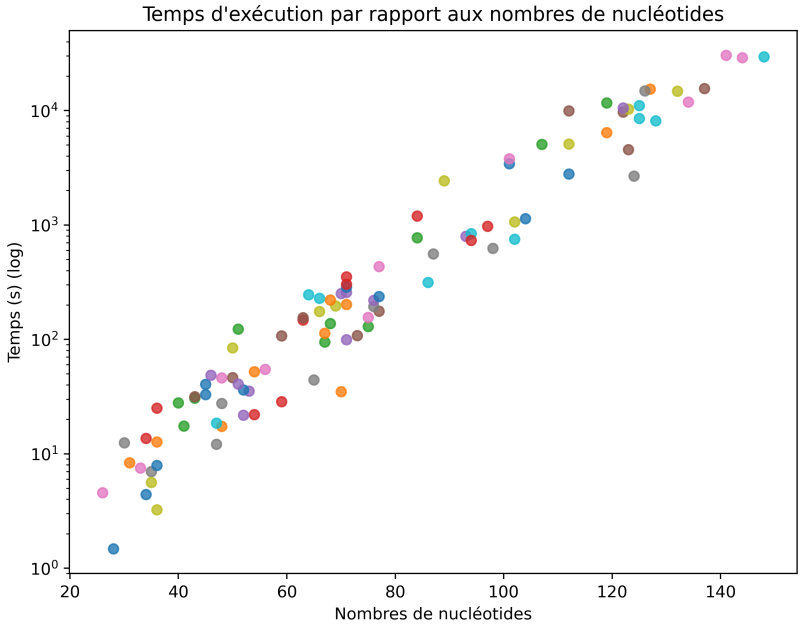


FIGURE 3.7 – **Temps d'exécution. Basé sur le nombre de nucléotides de la séquence lorsque** $\alpha = 0.1$. Le temps d'exécution croît de façon exponentielle vis-à-vis le nombre de nucléotides de la séquence.

CONCLUSION

Dans ce travail, un cadriciel de programmation en nombres entiers permettant la prédiction simultanée de la structure secondaire avec pseudo-nœuds et l'insertion de motifs structurels est présenté. L'implémentation dans RNA-MoIP est évaluée sur les 101 ARNs de pseudo-nœuds non redondantes dont la structure est connue.

Je montre que la combinaison de l'approche d'IPknot permettant de construire des structures incluant des pseudo-nœuds basées sur la matrice de probabilité des paires de bases obtenues par le modèle thermodynamique standard, implémentée dans le ViennaRNA, avec l'insertion de motifs structurels conservés connus, permet : (1) d'augmenter la précision de la prédiction de la structure secondaire avec des pseudo-nœuds (2) de générer des connaissances précises sur les interactions canoniques et de Wobble présentes à l'intérieur des motifs structurels, qui pourraient ne pas appartenir à la structure secondaire.

Deux limitations principales sont mises en évidence par notre travail. Premièrement, les motifs peuvent être insérés sur la base d'une correspondance parfaite entre les séquences. Des techniques probabilistes plus avancées, comme RMDetect (Cruz et Westhof, 2011), JAR3D (Roll *et al.*, 2016) ou BayesPairing (Sarrazin-Gendron *et al.*, 2020), permettraient d'intégrer un terme plus rigoureux dans la fonction objective, comme la correspondance des motifs avec une séquence altérée, augmentant la diversité et donc la gamme de structure prévisible. Deuxièmement, la base de données des motifs ne comprend que les boucles (c'est-à-dire les épingles à cheveux, les boucles intérieures, les multiboucles).

Les progrès de la biologie moléculaire structurelle repoussent les limites du modèle

du plus proche voisin. Alors que l'importance biologique des réseaux d'interactions non canoniques devient de plus en plus évidente, la capacité à les prédire est loin derrière. La programmation entière reste une direction prometteuse pour la détermination de la structure des ARNs en raison de la flexibilité et de leur formulation permettant d'aller au-delà des modèles thermodynamiques conventionnels. L'extension à des structures conservées plus complexes, comme les groupes de boucles en interaction et conservées contenant des pseudo-nœuds décrites dans Carnaval (Reinharz *et al.*, 2018; Soulé *et al.*, 2021) permettrait de profiter pleinement de la formulation entière et d'étendre la notion de prédiction des pseudo-nœuds à toutes les interactions non canoniques.

Beaucoup de techniques de prédiction de structure secondaire reposent sur l'utilisation d'alignement sur plusieurs séquences, disponibles sur certaines bases de données. Par ces données, cela permet de déterminer les sous-séquences les plus proéminentes, ce qui améliore le modèle énergique et raffine la prédiction. Il s'agit donc d'une optique intéressante qui aiderait à raffiner la recherche des paires de bases dans notre programme, lorsque ces alignements sont disponibles.

CHAPITRE IV

ANNEXE

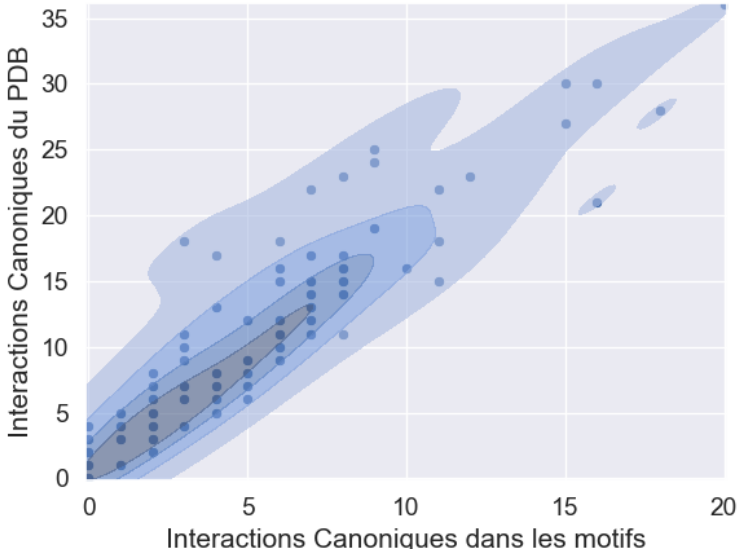


FIGURE 4.1 – **Interactions canoniques**. Distribution du nombre d'interactions canoniques qui sont observées aux emplacements des motifs insérés, à $\alpha = 0.1$. Sur l'axe des y, le nombre dans la structure réelle, sur l'axe des x, combien sont annotées dans le motif inséré.

RÉFÉRENCES

Akutsu, T. (2000). Dynamic programming algorithms for RNA secondary structure prediction with pseudoknots. *Discrete Appl. Math.*, *104*(1), 45–62. http://dx.doi.org/10.1016/S0166-218X(00)00186-4

Antczak, M., Zok, T., Popenda, M., Lukasiak, P., Adamiak, R. W., Blazewicz, J. et Szachniuk, M. (2014). RNApdbee—a webserver to derive secondary structures from pdb files of knotted and unknotted RNAs. *Nucleic Acids Research*, *42*(W1), W368–W372. http://dx.doi.org/10.1093/nar/gku330

Bellaousov, S. et Mathews, D. H. (2010). ProbKnot : Fast prediction of RNA secondary structure including pseudoknots. *RNA*, *16*(10), 1870–1880. http://dx.doi.org/10.1261/rna.2125310

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N. et Bourne, P. E. (2000). The protein data bank. *Nucleic Acids Res*, *28*(1), 235–42.

BGSU RNA (2022). Representative Sets of RNA 3D Structures . Récupéré le 15/04/2022 de http://rna.bgsu.edu/rna3dhub/nrlist/release/3.222/3.0A

Chen, H.-L., Condon, A. et Jabbari, H. (2009). An O(n5) Algorithm for MFE Prediction of Kissing Hairpins and 4-Chains in Nucleic Acids. *J. Comput. Biol.*, *16*(6), 803–815. http://dx.doi.org/10.1089/cmb.2008.0219

COIN-OR (2022). Cbc. Récupéré le 26/04/2022 de https://github.com/coin-or/Cbc

Cruz, J. A. et Westhof, E. (2011). Sequence-based identification of 3d structural modules in RNA with RMDetect. *Nature methods*, *8*(6), 513–519.

Darty, K., Denise, A. et Ponty, Y. (2009). VARNA : Interactive drawing and editing of the RNA secondary structure. *Bioinformatics*, *25*(15), 1974–1975. http://dx.doi.org/10.1093/bioinformatics/btp250

Djelloul, M. et Denise, A. (2008). Automated motif extraction and classification in RNA tertiary structures. *RNA*, *14*(12), 2489–2497.

Flamm, C., Wielach, J., Wolfinger, M. T., Badelt, S., Lorenz, R. et Hofacker, I. L. (2022). Caveats to Deep Learning Approaches to RNA Secondary Structure Prediction. *Front. Bioinform.*, *2*. http://dx.doi.org/10.3389/fbinf.2022.835422

Gianfrotta, C., Reinharz, V., Lespinet, O., Barth, D. et Denise, A. (2022). On the predictibility of A-minor motifs from their local contexts. *RNA Biol.*, *19*(1), 1208–1227. http://dx.doi.org/10.1080/15476286.2022.2144611

Gurobi Optimization, LLC (2022). Gurobi Optimizer Reference Manual, v9.5.0. Récupéré le 06/10/2022 de https://www.gurobi.com

ICEB, F. U. o. O. P. (2021). Python MIP Documentation — Python-MIP documentation. Récupéré le 15/04/2022 de https://docs.python-mip.com/en/latest/index.html

Leontis, N. B., Lescoute, A. et Westhof, E. (2006). The building blocks and motifs of RNA architecture. *Current opinion in structural biology*, *16*(3), 279–287.

Leontis, N. B. et Westhof, E. (2001). Geometric nomenclature and classification of RNA base pairs. *Rna*, *7*(4), 499–512.

Lorenz, R., Bernhart, S. H., Höner zu Siederdissen, C., Tafer, H., Flamm, C., Stadler, P. F. et Hofacker, I. L. (2011). ViennaRNA Package 2.0. *Algorithms for molecular biology*, *6*(1), 1–14.

McCaskill, J. S. (1990). The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, *29*(6-7), 1105–1119. http://dx.doi.org/10.1002/bip.360290621

Miao, Z., Adamiak, R. W., Antczak, M., Boniecki, M. J., Bujnicki, J., Chen, S.-J., Cheng, C. Y., Cheng, Y., Chou, F.-C., Das, R. *et al.* (2020). RNA-Puzzles Round IV : 3D structure predictions of four ribozymes and two aptamers. *RNA*, *26*(8), 982–995.

OR-Tools (2021). OR-Tools | Google Developers. [Online ; accessed 18. Oct. 2022]. Récupéré le 18/10/2022 de https://developers.google.com/optimization

Papadimitriou, C. H. (1981). On the complexity of integer programming. *Journal of the ACM (JACM)*, *28*(4), 765–768.

Reinharz, V., Major, F. et Waldispühl, J. (2012). Towards 3D structure prediction of large RNA molecules : an integer programming framework to

insert local 3D motifs in RNA secondary structure. *Bioinformatics*, *28*(12), i207–i214.

Reinharz, V., Soulé, A., Westhof, E., Waldispühl, J. et Denise, A. (2018). Mining for recurrent long-range interactions in RNA structures reveals embedded hierarchies in network families. *Nucleic acids research*, *46*(8), 3841–3851.

Reuter, J. S. et Mathews, D. H. (2010). RNAstructure : software for RNA secondary structure prediction and analysis. *BMC bioinformatics*, *11*(1), 1–9.

Rivas, E. et Eddy, S. R. (1999). A dynamic programming algorithm for RNA structure prediction including pseudoknots. *J. Mol. Biol.*, *285*(5), 2053–2068. http://dx.doi.org/10.1006/jmbi.1998.2436

Roll, J., Zirbel, C. L., Sweeney, B., Petrov, A. I. et Leontis, N. (2016). JAR3D Webserver : Scoring and aligning RNA loop sequences to known 3D motifs. *Nucleic acids research*, *44*(W1), W320–W327.

Sarrazin-Gendron, R., Yao, H.-T., Reinharz, V., Oliver, C. G., Ponty, Y. et Waldispühl, J. (2020). Stochastic sampling of structural contexts improves the scalability and accuracy of RNA 3D module identification. Dans *International Conference on Research in Computational Molecular Biology*, 186–201. Springer.

Sato, K., Akiyama, M. et Sakakibara, Y. (2021). RNA secondary structure prediction using deep learning with thermodynamic integration. *Nat. Commun.*, *12*(941), 1–9. http://dx.doi.org/10.1038/s41467-021-21194-4

Sato, K. et Kato, Y. (2022). Prediction of RNA secondary structure including pseudoknots for long sequences. *Briefings in Bioinformatics*, *23*(1), bbab395.

Sato, K., Kato, Y., Hamada, M., Akutsu, T. et Asai, K. (2011). IPknot : fast and accurate prediction of RNA secondary structures with pseudoknots using integer programming. *Bioinformatics*, *27*(13), i85–93. http://dx.doi.org/10.1093/bioinformatics/btr215

Smit, S., Rother, K., Heringa, J. et Knight, R. (2008). From knotted to nested RNA structures : a variety of computational methods for pseudoknot removal. *RNA*, *14*(3), 410–416.

Soulé, A., Reinharz, V., Sarrazin-Gendron, R., Denise, A. et Waldispühl, J. (2021). Finding recurrent RNA structural networks with fast maximal common subgraphs of edge-colored graphs. *PLoS computational biology*,

17(5), e1008990.

Szikszai, M., Wise, M., Datta, A., Ward, M. et Mathews, D. H. (2022). Deep learning models for RNA secondary structure prediction (probably) do not generalize across families. *Bioinformatics*, *38*(16), 3892–3899. http://dx.doi.org/10.1093/bioinformatics/btac415

Tinoco, Jr., I. et Bustamante, C. (1999). How RNA folds. *J. Mol. Biol.*, *293*(2), 271–281. http://dx.doi.org/10.1006/jmbi.1999.3001

Turner, D. H. et Mathews, D. H. (2010). NNDB : the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure. *Nucleic Acids Res.*, *38*(1), D280–D282. http://dx.doi.org/10.1093/nar/gkp892

van Batenburg, F. H., Gultyaev, A. P. et Pleij, C. W. (2001). PseudoBase : structural information on RNA pseudoknots. *Nucleic Acids Res.*, *29*(1), 194–195. http://dx.doi.org/10.1093/nar/29.1.194

Wang, F., Zuroske, T. et Watts, J. K. (2020). RNA therapeutics on the rise. *Nat Rev Drug Discov*, *19*(7), 441–442.

Watkins, A. M., Rangan, R. et Das, R. (2020). FARFAR2 : improved de novo rosetta prediction of complex global RNA folds. *Structure*, *28*(8), 963–976.

Yao, J., Reinharz, V., Major, F. et Waldispühl, J. (2017). RNA-MoIP : prediction of RNA secondary structure and local 3D motifs from sequence data. *Nucleic acids research*, *45*(W1), W440–W444.

Yu, A.-M., Choi, Y. H. et Tu, M.-J. (2020). RNA drugs and RNA targets for small molecules : principles, progress, and challenges. *Pharmacological reviews*, *72*(4), 862–898.

Zok, T., Antczak, M., Zurkowski, M., Popenda, M., Blazewicz, J., Adamiak, R. W. et Szachniuk, M. (2018). RNApdbee 2.0 : multifunctional tool for RNA structure annotation. *Nucleic Acids Research*, *46*(W1), W30–W35. http://dx.doi.org/10.1093/nar/gky314