Université de Montréal

Modélisation des réseaux de régulation de l'expression des gènes par les microARN

Par
Guillaume Poirier-Morency

Département d'informatique et de recherche opérationelle
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de Maîtrise ès science en informatique, option biologie computationnelle

Septembre 2020

Université de Montréal

Département d'informatique et de recherche opérationnelle

Institut de recherche en immunologie et cancérologie

Ce mémoire intitulé

Modélisation des réseaux de régulation de l'expression des gènes par les microARN

Présenté par

Guillaume Poirier-Morency

À été évalué par un jury composé des personnes suivantes

Sylvie Hamel

Président-rapporteur

François Major

Directeur de recherche

Nadia El-Mabrouk

Membre du jury

Résumé et mots-clés

Résumé

Les microARN sont de petits ARN non codants d'environ 22 nucléotides impliqués dans la régulation de l'expression des gènes. Ils ciblent les régions complémentaires des molécules d'ARN messagers que ces gènes codent et ajustent leurs niveaux de traduction en protéines en fonction des besoins de la cellule.

En s'attachant à leurs cibles par complémentarité partielle de leurs séquences, ces deux groupes de molécules d'ARN compétitionnent activement pour former des interactions régulatrices. Par conséquent, prédire quantitativement les concentrations d'équilibres des duplexes formés est une tâche qui doit prendre un compte plusieurs facteurs dont l'affinité pour l'hybridation, la capacité à catalyser la cible, la coopérativité et l'accessibilité de l'ARN cible.

Dans le modèle que nous proposons, miRBooking 2.0, chaque interaction possible entre un microARN et un site sur un ARN cible pour former un duplexe est caractérisée par une réaction enzymatique. Une réaction de ce type opère en deux phases : une formation réversible d'un complexe enzyme-substrat, le duplexe microARN-ARN, suivie d'une conversion irréversible du substrat en produit, un ARN cible dégradé, et de la restitution l'enzyme qui pourra participer à une nouvelle réaction.

Nous montrons que l'état stationnaire de ce système, qui peut comporter jusqu'à 10 millions d'équations en pratique, est unique et son jacobien possède un très petit nombre de valeurs non-nulles, permettant sa résolution efficace à l'aide d'un solveur linéaire épars. Cette solution nous permet de caractériser précisément ce mécanisme de régulation et d'étudier le rôle des microARN dans un contexte cellulaire donné. Les prédictions obtenues sur un modèle de cellule HeLa corrèlent significativement avec un ensemble de données obtenu expérimentalement et permettent d'expliquer remarquablement les effets de seuil d'expression des gènes. En utilisant ces prédictions comme condition initiale et une méthode d'intégration numérique, nous simulons en temps réel la réponse du système aux changements de conditions expérimentales.

Nous appliquons ce modèle pour cibler des éléments impliqués dans la transition épithélio-mésenchymateuse (EMT), un mécanisme biologique permettant aux cellules

d'acquérir une mobilité essentielle pour proliférer. En identifiant des éléments transcrits différentiellement entre les conditions épithéliale et mésenchymateuse, nous concevons des microARN synthétiques spécifiques pour interférer avec cette transition. Pour ce faire, nous proposons une méthode basée sur une recherche gloutonne parallèle pour rechercher efficacement l'espace de la séquence du microARN et présentons des résultats préliminaires sur des marqueurs connus de l'EMT.

Mots-clés : microARN, modélisation mathématique, équation de Michaelis-Menten, recherche de zéro en plusieurs dimensions, intégration numérique, analyses de données de séquençage à haut débit.

Résumé et mots-clés en anglais

Abstract

MicroRNAs are small non-coding RNAs of approximately 22 nucleotide long involved in the regulation of gene expression. They target complementary regions to the RNA transcripts molecules that these genes encode and adjust the concentration according to the needs of the cell.

As microRNAs and their RNA targets binds each other with imperfect complementarity, these two groups actively compete to form regulatory interactions. Consequently, attempting to quantitatively predict their equilibrium concentrations is a task that must take several factors into account, including the affinity for hybridization, the ability to catalyze the target, cooperation, and RNA accessibility.

In the model we propose, miRBooking 2.0, each possible interaction between a microRNA and a binding site on a target RNA is characterized by an enzymatic reaction. A reaction of this type operates in two phases: a reversible formation of an enzyme-substrate complex, the microRNA-RNA duplex, and an irreversible conversion of the substrate in an RNA degradation product that restores the enzyme which can subsequently participate to other reactions.

We show that the stationary state of this system, which can include up to 10 million equations in practice, has a very shallow Jacobian, allowing its efficient resolution using a sparse linear solver. This solution allows us to characterize precisely the mechanism of regulation and to study the role of microRNAs in a given cellular context. Predictions obtained on a HeLa S3 cell model correlate significantly with a set of experimental data obtained experimentally and can remarkably explain the expression threshold effects of genes. Using this solution as an initial condition and an explicit method of numerical integration, we simulate in real time the response of the system to changes of experimental conditions.

We apply this model to target elements involved in the Epithelio-Mesenchymal Transition (EMT), an important mechanism of tumours proliferation. By identifying differentially expressed elements between the two conditions, we design synthetic microRNAs to interfere with the transition. To do so, we propose a method based on a

parallel greedy best-first search to efficiently crawl the sequence space of the microRNA and present preliminary results on known EMT markers.

Keywords: microRNA, mathematical modelling, Michaelis-Menten enzyme kinetics, multidimensional root-finding, numerical integration, high-throughput sequencing data analysis.

Table des matières

Table des figures

Liste des tableaux

Liste des sigles et abréviations

AGO protéine de la famille Argonaute

ADN acide désoxyribonucléique

ARN acide ribonucléique

ARNm ARN messager codant pour une protéine

ARNnc ARN non-codant

CGI *common gateway interface*

CLIP *cross-linking* suivi d'une immunoprécipitation

EMT transition epithélio-mésenchymateuse

MRE motif d'ARN cible reconnu par un complexe RISC

PCR réaction en chaîne par polymérase

RISC complexe composé d'une protéine de la famille Argonaute et d'un ARN guide

RNA-Seq méthode de séquençage à haut débit des ARN

SMART petit ARN interférant synthétique

UTR région non traduite de l'ARNm

dsRNA petit ARN double-brin pouvant être chargé dans un complexe RISC

microARN petit ARN endogène interférant d'environ 22 nucléotides

nd. non disponible

Par souci de compacité, le tableau 1 présente une brève revue des notations utilisé au cours de ce mémoire.

Tab. 1 : Tableau résumé des notations utilisées et leurs unités correspondantes.

Notation	Description	Unité
$m :: t, p$	Duplex formé d'un microARN et d'un MRE	N/A
$\Delta G_{m::t,p}$	Énergie libre du duplex formé du microARN m à la position p de la cible t	kJ/mol
$[X]$	Concentration de X	pM
$[E_m]$	Concentration du microARN m	pM
$[S_t]$	Concentration du substrat t	pM
$[S_{t,p}]$	Concentration de la position p du substrat t	pM
$[E_m S_{t,p}]$	Concentration du complexe formé du microARN m à la position p du substrat t	pM
$[P_t]$	Concentration du produit t	pM
k	Constante de taux	selon la réaction
K_{eq}	Constante d'équilibre	selon la réaction
K_d	Constante de dissociation	selon la réaction
$\Pr[X]$	Probabilité de l'événement X	nd.
$\sum_m$	Somme sur tous les microARN du système	nd.
$\sum_t$	Somme sur tous les ARN cibles du système	nd.
$\sum_{p \in t}$	Somme sur toutes les positions d'un ARN	nd.

Dédicace

À Carol, Julien Mateo et Eva Simone, mes amours.

Remerciements

Je n'aurais pu réaliser ce travail sans le support de ma famille et de mon équipe au laboratoire du Dr. François Major qui m'a fourni tout le contexte biologique nécessaire.

En premier lieu, à Olivier Mailhot qui a suggéré l'idée d'utiliser un modèle aux équations différentielles partielles sur une version simplifiée du problème étudié dans ce mémoire.

En particulier, je souhaite remercier Albert Feghaly de m'avoir particulièrement bien complémenté pour la partie bio-informatique et guidé pour les analyses RNA-Seq. Il a également été une source inépuisable d'inspiration qui, à travers maintes discussions, a guidé de manière effective mon investigation.

Aussi, j'offre des remerciements spéciaux à Jeremie Zumer pour les discussions fructueuses sur l'aspect mathématique du modèle de miRBooking 2.0. Il a notamment trouvé une erreur dans une preuve qui a mené à une simplification importante et un gain substantiel en coût de calcul.

Finalement, je souhaite remercier les auteurs et contributeurs des projets de logiciels libres suivants :

- GLib et GObject ;
- LaTeX, Pandoc et toutes les bibliothèques utilisées pour produire ce mémoire ;
- Meson ;
- Python et les bibliothèques autograd, NumPy, SciPy, Seaborn et Pandas.

Chapitre 1

Introduction

1.1 Biologie des acides ribonucléiques

La cellule est un système complexe et hétérogène subdivisée en compartiments séparés par des membranes et peuplée d'organites lui permettant de réaliser ses fonctions : croître, produire de l'énergie, se reproduire, etc. Tout ce système est principalement régi par l'unité héréditaire : le génome. Dans la cellule eucaryote, ce génome est constitué d'une ou plusieurs longues molécules d'acide désoxyribonucléique (ADN) nommés chromosomes qui sont localisés dans le noyau. L'ADN est une longue séquence polymérique dont l'alphabet est constitué de quatre lettres nommées nucléotides. Afin de formuler les instructions pour réaliser ses fonctions biologiques, certaines régions de l'ADN encodent des gènes qui sont transcrits en acides ribonucléiques (ARN) complémentaires par la polymérase.

Les ARN agissent généralement comme messagers (ARNm) pour des protéines spécifiques qui seront traduits par des ribosomes, mais ils n'y sont pas limités : certains ARN sont à eux seuls fonctionnels comme les ARNm présentant des riboswitches et les ARN de transfert (ARNt) et d'autres se retrouvent assemblés dans des complexes avec des protéines tels que les télomérases et les complexes de répression induite par un ARN guide (RISC).

Pour la cellule, l'abondance des différentes molécules qu'elle produit, importe ou exporte de son environnement est critique à la réalisation de ses fonctions biologiques et cette dernière possède une panoplie de mécanismes pour réaliser cet objectif. L'ARN ne fait pas exception à la règle.

Il existe donc une panoplie de mécanismes conservés pour réguler l'abondance des ARN dont les facteurs de transcription et les boucles d'autorégulation. L'expression des gènes est régulée à plusieurs niveaux : au niveau génomique par des facteurs de transcription qui activent ou désactivent certaines régions des chromosomes, au niveau post-transcriptionnel et post-traductionnel par des enzymes qui dégradent respectivement les molécules d'ARN et

les protéines. Une boucle d'autorégulation peut agir sur plusieurs niveaux simultanément : la protéine associée à un gène peut activer la transcription d'un microARN qui réprime son ARNm, créant ainsi un niveau d'expression spécifique et stable.

Ce mémoire porte spécifiquement sur le mécanisme post-transcriptionnel de la régulation des gènes médiée par les microARN, c'est-à-dire une fois que le message est transmis du noyau sous forme d'un ARNm et avant qu'il soit traduit en protéine par un ribosome.

1.2 Historique des microARN

Les microARN ont été découverts en 1993 chez *C. elegans* par Lee, Feinbaum et Ambros [1] qui étudiaient le rôle du gène lin-4 dans le développement post-embryonnaire de cet organisme. Ils ont réalisé que ce gène encodait deux petites séquences d'ARN de 61 et 22 nucléotides complémentaires à la région 3'UTR de l'ARN messager de lin-4. Ces deux types de séquences, aujourd'hui connues et retrouvées parmi pratiquement tous les organismes vivants, sont les transcrits précurseur et mature des microARN. Ils servent précisément à interférer avec l'expression des gènes auxquels ils sont complémentaires.

Quelques années plus tard, un autre microARN, let-7, est identifié et caractérisé par Reinhart et al. [2]. Il en fallut peu pour que le nombre de séquences identifiées explose : miRBase répertorie aujourd'hui 38,589 séquences de précurseurs produisant 48,860 microARN matures uniques [3].

En 2009, Chi et al. [4] réussissent à précipiter les complexes RISC et à séquencer leurs cibles à l'aide de la méthode HITS-CLIP, permettant ainsi une analyse très précise des sites d'hybridation des microARN à grande échelle.

En 2015, Agarwal et al. [5] dévoilent TargetScan 7.0 qui deviendra assez rapidement la méthode la plus utilisée pour les prédictions de cibles de microARN en mettant de l'avant deux critères : la complémentarité des nucléotides 2 à 8 de l'ARN guide constituant son amorce et la conservation des régions du 3'UTR de l'ARNm.

Puis en 2016, Broughton et al. [6] présente la variante iCLIP permettant d'identifier simultanément le microARN et la cible en les liguant ensemble et d'établir une famille de modèle d'interactions possibles.

Récemment, Wang et al. [7] ont publié une technique permettant de co-séquencer les ARN messager et les microARN d'une seule cellule et d'obtenir le portrait très précis des abondances relatives.

En parallèle aux efforts menés dans les analyses à haut débits, Zamore et al. [8], Wee et al. [9], Salomon et al. [10], Becker et al. [11] et plusieurs autres auteurs ont contri-

bué activement à établir une perspective mécaniste et biochimique des interactions entre le complexe RISC et un motif reconnu sur l'ARN cible dénommé MRE.

Plusieurs approches pour étudier ces interactions ont donc évolué parallèlement

1.3 Biogenèse du microARN

Le microARN provient originellement du génome, encodé de manière quasi palindromique et transcrit par la polymérase d'ARN II pour obtenir un microARN primaire. Une fois transcrit, il se replie sur lui-même afin d'adopter une conformation de tête d'épingle lui permettant de se lier à Drosha, une ribonucléase qui va couper les brins d'ARN qui se balancent librement. On parlera alors d'un pré-microARN qui est exporté du noyau cellulaire vers le cytoplasme par la protéine Exportin 5.

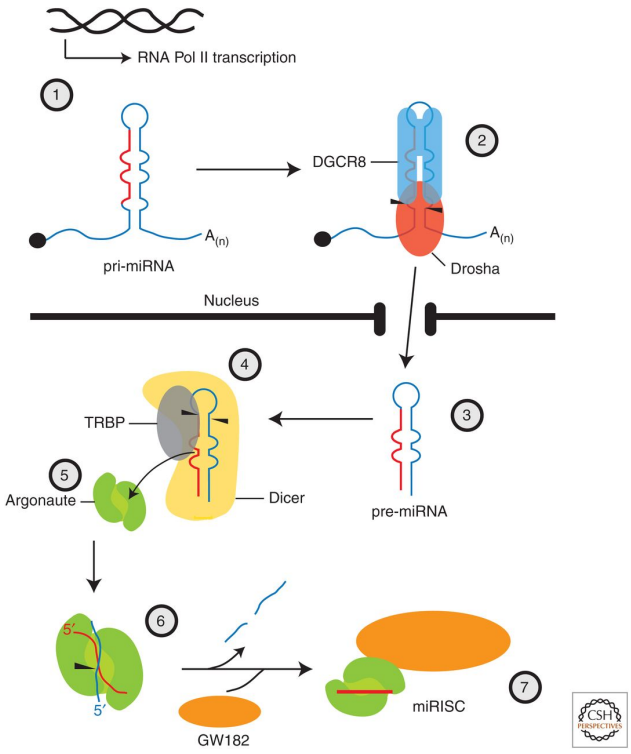


FIG. 1.1 : Résumé des différentes étapes constituant la la biogenèse du microARN mature. Figure provenant de DUCHAINE et FABIAN [12].

Le pré-microARN se lie ensuite à Drosha et Argonaute qui vont couper la boucle de la tête d'épingle. La section linéaire sera séparée en deux microARN matures : l'un sera chargé dans la protéine Argonaute pour constituer le complexe RISC et l'autre sera dégradé. Le choix dépend principalement d'éléments de la structure secondaire et de la stabilité de chaque brin [13].

Une fois chargé dans le complexe RISC, il peut diffuser librement dans le cytoplasme afin de réaliser sa fonction régulatrice.

Il existe également des voies non canoniques pour la biogenèse de certains microARN matures indépendantes de Drosha [14] et Dicer [15, 16]. Néanmoins, le résultat final reste le même : un complexe RISC prêt à réguler ses cibles.

1.4 Interactions entre les microARN et leurs cibles

Le complexe RISC interagit avec un ARN cible par complémentarité imparfaite de son microARN guide. Les sites fonctionnels sont typiquement situés dans la région 3'UTR des ARN messagers [17] et conservés [18] pour maintenir la fonction régulatrice. L'ensemble des interactions entre microARN et leurs ARN cibles constitue le microtargetome.

La spécificité du guide est principalement attribuée à la complémentarité de son amorce qui s'étend des nucléotides 2 à 8 avec son substrat [18] qui sert de mécanisme de reconnaissance au complexe. À ces nucléotides peuvent s'accompagner des appariements supplémentaires du côté 3' du guide qui stabilisent le complexe. Dans certains rares cas, le microARN est suffisamment complémentaire pour que la région centrale s'hybride, menant au clivage du substrat par le domaine PIWI d'Argonaute. La complexité de ce mécanisme s'explique par le fait que le complexe cherche à éviter de se compromettre inutilement sur des sites non-spécifiques en subdivisant ce processus en plusieurs étapes réversibles [20].

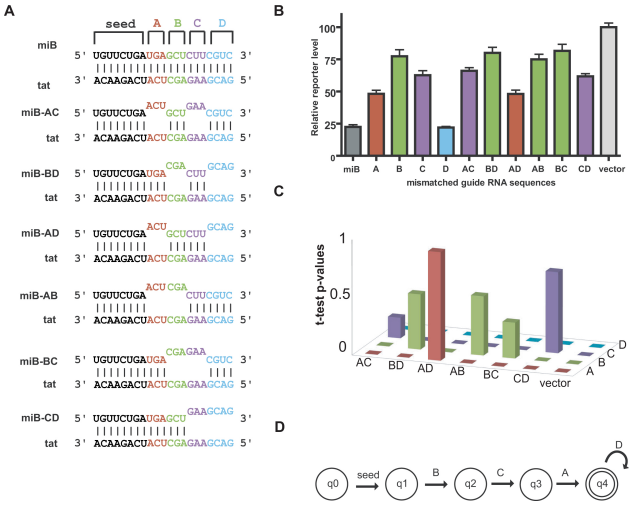


FIG. 1.2 : Modèle séquentiel proposé par [19] pour faire le pont entre les changements conformationnels du complexe RISC et les appariements des nucléotides du microARN guide. L'automate décrit les états et transitions possibles pour les appariements : le microARN lie d'abort les nucléotides de son empreinte et par la suite ceux des boîtes B, C et D. Figure provenant de YAN et al. [19].

Le modèle mécaniste proposé par YAN et al. [19] du laboratoire Major suggère que des régions spécifiques du microARN guide s'hybrident à sa cible en suivant une séquence de changements conformationnels du complexe RISC (voir figure 1.2). Il est également intéressant de noter que le complexe tolère assez bien des structures secondaires de type *bulge* tant du côté de l'ARN cible que du microARN en subdivisant le guide en deux chambres autours du site de clivage [21].

1.5 miRBooking 1.0

En 2015, WEILL et al. [22] publient la première mouture de miRBooking, un modèle basé sur la résolution du problème de mariage stable avec l'algorithme de GALE et SHAPLEY [23] pour obtenir les associations préférentielles entre un ensemble de complexes RISC et leurs

cibles.

Un exemple d'un problème de mariage stable est celui d'un ensemble de patients que l'on souhaite assigner à un ensemble de médecins traitants. De la perspective de chaque patient, les médecins sont ordonnés en fonction de préférences personnelles, pertinence de la spécialité, etc. L'algorithme de Gale et Shapley [23] obtient une solution telle qu'aucun patient n'aurait intérêt à échanger son médecin avec un autre patient, la définition même d'un mariage stable. Par contre, elle ne garantira pas qu'une paire de médecins n'aurait pas avantage à échanger leurs patients.

Tant qu'il existe un patient p non assigné à un médecin, on prend ce patient et on considère le prochain médecin m en ordre de préférences décroissantes auquel le patient n'a pas encore été proposé et on lui propose en appliquant la procédure suivante :

1. si le médecin m est libre, il est attribué au patient p, la proposition pour p est un succès ;
2. si le médecin m est attribué à un patient p' de préférence inférieure à p, on l'attribue au patient p et l'autre patient p' devient non assigné ;
3. sinon p' conserve son médecin m, la proposition pour p est un échec.

Éventuellement, tous les patients sont assignés puisqu'à chaque étape le nombre de couples médecin-patient s'étant proposé croît strictement et cette quantité est bornée par $O(n^2)$. La solution obtenue constitue un ensemble de mariages stables de la perspective des patients.

miRBooking procède de manière similaire en discrétisant tout d'abord les quantités de complexes RISC et leurs cibles en les assignant par la suite à l'aide d'une matrice de préférences. Comme il serait très coûteux d'appliquer directement l'algorithme de Gale et Shapley [23], ce dernier étant situé dans l'ordre de $O(n^2)$, les quantités de microARN à attribuer à chaque cible sont approximées en une seule passe d'assignations préférentielles.

Le modèle est simple et reflète assez bien la globalité du système. La solution obtenue est physiquement réaliste puisqu'elle ne contrevient pas au principe de conservation de la masse. Par contre, il possède quelques limitations que la méthode proposée dans ce mémoire tente d'adresser :

- on suppose que les microARN peuvent choisir leurs cibles à même titre qu'un patient pourrait choisir son médecin ;
- on ne considère pas les valeurs réelles des affinités chimiques, mais plutôt une transformation en probabilités des énergies libres ;
- les concentrations d'origine sont peu importantes pour les microARN faiblement exprimés puisque les quantités sont discrétisées ;

- certains aspects du modèle comme la disponibilité d'un site est approximative et nous proposons ici une formulation exacte ;
- la solution obtenue ne constitue pas un équilibre chimique.

Dans les prochaines sections, nous traiterons de concepts qui constituent les fondements de la méthode miRBooking 2.0 qui sera présentée dans les premier et second chapitres cet ouvrage. En particulier, nous montrerons comment reformuler ce problème à l'aide d'un cadre biochimique faisant intervenir les concentrations des espèces en interaction et les constantes d'affinités et d'efficacités enzymatiques de chaque complexe RISC::MRE formé. Le reste de l'introduction et les chapitres qui suivent constitue une contribution originale de l'auteur.

1.6 Réactions chimiques : loi d'action de masse

Le modèle mathématique le plus généralement utilisé pour les réactions chimiques est dérivé de la loi d'action de masse. Elle stipule que le taux de progression d'une réaction chimique est proportionnel aux concentrations libres des réactifs qu'elle fait intervenir.

Par exemple, si S réagit pour former un produit P, le taux auquel la concentration de P, notée $[P]$, varie est proportionnel à la concentration de S à un instant donné.

$$\frac{\partial[P]}{\partial t} \propto [S] \tag{1.1}$$

Cette relation de proportionnalité implique qu'il existe une certaine constante de taux k qui caractérise notre réaction.

$$\frac{\partial[P]}{\partial t} = k[S] \tag{1.2}$$

Le taux d'une réaction faisant intervenir plusieurs réactifs est proportionnel aux concentrations libres de chacun d'entre eux :

$$\frac{\partial[Q]}{\partial t} \propto [S_1] \times [S_2] \times ... \times [S_n] \tag{1.3}$$

$$\implies \frac{\partial[Q]}{\partial t} = k[S_1] \times [S_2] \times ... \times [S_n] \tag{1.4}$$

Pour résoudre ce type d'équation, on note que $[S]$ est fonction du temps et par conséquent, on peut écrire l'équation différentielle ordinaire suivante :

$$\frac{\partial[P](t)}{\partial t} - k[S](t) = 0 \tag{1.5}$$

$$\frac{\partial[P](t)}{\partial t} + k[P](t) - k[S]_0 = 0 \qquad \text{par conservation, } [S] = [S]_0 - [P] \tag{1.6}$$

Cette équation différentielle ordinaire est linéaire et non-homogène et possède une solution de forme exponentielle $[P](t) = [S]_0(1 - e^{-kt})$ qu'on peut vérifier par substitution dans l'équation 1.6.

1.7 Réactions réversibles

En général, les réactions chimiques sont réversibles : il existe donc un processus complémentaire qui effectue la réaction inverse. Lorsqu'on incorpore cette notion, cela nous permet de parler d'équilibre chimique puisqu'il existera un point où la réaction et son inverse se produiront au même rythme.

Admettons qu'une réaction produit spontanément $[P]$ à partir de $[S]$ et qu'une réaction complémentaire effectue le travail inverse à partir de $[P]$.

$$[\mathrm{S}] \underset{k_2}{\overset{k_1}{\rightleftharpoons}} [\mathrm{P}]$$

La paire d'équations suivantes détermine le comportement du système :

$$\frac{\partial[P]}{\partial t} = k_1[S] - k_2[P] \tag{1.7}$$

$$\frac{\partial[S]}{\partial t} = k_2[P] - k_1[S] \tag{1.8}$$

L'équilibre est atteint lorsque $\frac{\partial[S]}{\partial t} = \frac{\partial[P]}{\partial t} = 0$ puisque les concentrations des deux espèces S et P deviennent stables. Cela ne signifie pas pour autant qu'aucune réaction n'a lieu, mais qu'il y a autant d'espèces produites que détruites.

On note que l'équilibre d'une réaction entraîne celle de l'autre, ce qui nous permet de considérer seulement la première équation :

$$\begin{aligned} \frac{\partial[P]}{\partial t} &= k_1[S] - k_2[P] \\ \implies 0 &= k_1[S] - k_2[P] \\ \implies \frac{[P]}{[S]} &= \frac{k_1}{k_2} = K_{eq} \end{aligned}$$

Le rapport entre k_1 et k_2 est utilisé pour calculer la constante d'équilibre de la réaction notée K_{eq} puisqu'il correspond au ratio de concentration des réactifs et des produits à l'équilibre.

L'inverse de ce rapport est la constante de dissociation notée K_d. Cette dernière est liée à l'énergie libre du système via l'équation de Gibbs : $K_d = e^{\Delta G/RT}$. Nous verrons plus tard que ce lien est très important, car il permet de faire le pont entre l'énergie de repliement des ARN et macromolécules et la notion d'équilibre chimique.

1.8 Réactions enzymatiques : équation de Michaelis-Menten

Une réaction enzymatique fait intervenir un enzyme $[E]$ et un substrat $[S]$ dans un processus de formation d'un complexe $[ES]$ et d'un produit $[P]$. La particularité de cette réaction est que l'enzyme est finalement récupéré et réutilisé pour d'autres réactions tandis que le substrat, converti en produit, est irréversiblement transformé.

$$\text{[E]} + \text{[S]} \underset{k_f}{\overset{k_r}{\rightleftharpoons}} \text{[ES]} \xrightarrow{k_{cat}} \text{[E]} + \text{[P]}$$

La progression d'une réaction chimique est typiquement guidée par la loi d'action de masse, qui établit des taux de variation proportionnels aux concentrations en interaction. Une réaction qui se conforme au modèle enzymatique de Michaelis-Menten [24] se caractérise par l'ensemble des équations différentielles suivantes :

$$\frac{\partial[E]}{\partial t} = -k_f[E][S] + k_r[ES] + k_{cat}[ES]$$
$$\frac{\partial[S]}{\partial t} = -k_f[E][S] + k_r[ES]$$
$$\frac{\partial[ES]}{\partial t} = k_f[E][S] - k_r[ES] - k_{cat}[ES]$$
$$\frac{\partial[P]}{\partial t} = -k_{cat}[ES]$$

En posant $\frac{\partial[ES]}{\partial t} = 0$, on obtient la constante d'efficacité enzymatique K_m qui caractérise le rapport entre les concentrations d'enzymes et substrats libres relatifs aux complexes formés lorsque la réaction est stationnaire[1].

$$K_m = \frac{[E][S]}{[ES]} = \frac{k_r + k_{cat}}{k_f} \tag{1.9}$$

Notablement, l'activité du complexe RISC est enzymatique [8] puisque le complexe s'associe et se dissocie librement à son ARN substrat et peut, en atteignant la conformation nécessaire, couper son substrat pour ensuite être réutilisé pour une autre réaction (voir figure 1.3).

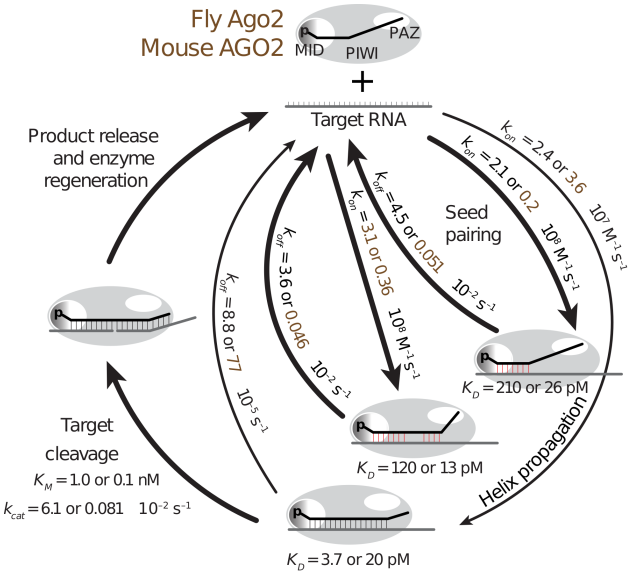


FIG. 1.3 : Les interactions entre les microARN et leurs cibles constituent un exemple d'application de l'équation de Michaelis-Menten. Figure provenant de WEE et al. [9].

1.9 Matrices éparses

Une matrice éparse, ou matrice creuse est une matrice contenant un nombre important de zéros. Il est avantageux pour ce type de données d'adopter un encodage plus efficace que celui donné par un tableau multidimensionnel.

[1] On sous-entend que le substrat afflue dans le système au même rythme qu'il est transformé en produit.

Encodage épars de Yale

L'encodage épars de Yale utilise trois listes pour encoder les valeurs non-nulles d'une matrice éparse. La première sert à définir les intervalles de stockage des valeurs non nulles d'une rangée dans les deux autres listes. La deuxième liste donne les indices des colonnes et la troisième donne les valeurs numériques correspondantes.

Par exemple, on représente la matrice suivante :

$$M = \begin{bmatrix} 1 & 0 & 2 & 3 \\ 0 & 4 & 5 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 6 & 0 \end{bmatrix} \tag{1.10}$$

à l'aide des trois listes suivantes :

```
rowptr = [0, 3, 5, 5, 6]
colind = [0, 2, 3, 1, 2, 2]
data = [1, 2, 3, 4, 5, 6]
```

Pour récupérer les indices de colonnes et les valeurs d'une rangée `i` donnée, `rowptr` donne directement les intervalles à chercher dans les deux autres tableaux :

```
j = colind[rowptr[i]:rowptr[i+1]]
x = data[rowptr[i]:rowptr[i+1]]
```

Pour la première rangée, `rowptr` donne l'intervalle $[0, 3[$ donnant accès aux indices et valeurs suivantes dans `colind` et `data` :

```
j = [0, 2, 3]
x = [1, 2 ,3]
```

et permettent de reconstruire la première rangée $\begin{bmatrix} 1 & 0 & 2 & 3 \end{bmatrix}$ de la matrice.

Lorsque les sous-intervalles de colonnes sont maintenues en ordre croissant, une recherche binaire peut être effectuée dans la liste d'indices des colonnes pour trouver la valeur correspondante en $O(\log n)$.

Il existe une multitude d'autres façons d'encoder des matrices éparses telle que par une liste de coordonnées et par blocs, mais seulement cet encodage sera considéré pour les applications faites dans ce mémoire.

1.10 Motivation

Le travail présenté est assez complexe et il est important de discuter des motivations qui m'ont poussé à l'entreprendre. Lorsque j'ai débuté ma maîtrise, le laboratoire de François venait de publier un article sur la première mouture de miRBooking [22]. Durant les années qui ont suivi, plusieurs articles ont été publiés fournissant des données et des mesures permettant de voir le problème d'un tout nouvel angle [10, 25, 26, 21]. On savait désormais l'influence de plusieurs facteurs sur la force de l'interaction du complex RISC en termes de constantes d'équilibre et catalytique. Toute cette nouvelle information ouvrait la porte à un modèle qui serait réaliste du point de vue biochimique et qu'on pourrait bâtir à partir des notions introduites ci-haut.

Au début, nous avons eu beaucoup de doutes sur la possibilité de trouver l'état d'équilibre d'un tel système étant donné le nombre d'équations à résoudre. Un collègue du laboratoire, Olivier Mailhot, avait déjà expérimenté avec des équations différentielles sur une version simplifiée du modèle sans trop de succès : l'intégration numérique était beaucoup trop coûteuse à appliquer. Une fois que j'avais terminé d'écrire les équations mathématiques qui décriront la dynamique du système, j'ai réalisé que son jacobien était quasiment nul. Euréka ! les méthodes de deuxième ordre telle que la méthode de Newton pouvait être appliquée de manière efficace à l'aide d'un solveur épars. On pouvait alors envisager résoudre des instances assez grandes. Quelques mois plus tard, nous étions en mesure de résoudre des systèmes de plus de 300,000 équations !

Au moment de rédiger ce document, la recherche s'est poursuivie et d'importants résultats ont été publiés dont celui de Becker et al. [11] qui ont mesuré les constantes pour plus de 40,000 interactions. Ils proposent également un modèle d'interaction permettant de prédire les valeurs K_d et k_{cat} que nous envisageons unifier avec notre modèle d'équilibre global.

Il est important de prendre en considération que le modèle proposé fait abstraction de la manière dont les constantes et les conditions initiales sont obtenues. La contribution principale de cet ouvrage est plutôt de démontrer qu'un modèle réaliste et efficace de régulation de l'expression des gènes par les microARN est possible. Malheureusement, cela entraîne une importante limitation pour le modèle puisque la précision de la solution dépend essentiellement de la qualité des estimations des conditions initiales et de notre capacité à modéliser précisément les facteurs qui influencent les propriétés biochimiques du complexe RISC avec sa cible, et nous sommes encore très loin de cet idéal.

Chapitre 2

miRBooking 2.0 : un modèle enzymatique pour le microtargetome

Les études sur les propriétés biochimiques du complexe RISC ont apporté beaucoup de lumière sur la nature quantitative de ses interactions. Haley et Zamore [27] ont montré que le complexe RISC agit comme un enzyme en présence d'ATP en se liant, dissociant et catalysant librement son ARN cible. D'autres auteurs ont investigué la spécificité du clivage [28] et le taux d'arrivée du complexe [26] à un site d'hybridation. Des modèles biophysiques ont déjà été proposés [29] et ces interactions ont été étudiées sous un modèle de concentration d'équilibre [30].

Des méthodes à haut débit du genre HITS-CLIP [4], PAR-CLIP et iCLIP [6] offre des approches très fructueuses pour quantifier et localiser ces interactions, mais incombent un biais intrinsèque puisque le *cross-linking* lie irréversiblement la protéine sur sa cible et le séquençage introduit des artéfacts nuisibles [31].

En 2015, la version originale de miRBooking [22] a été publiée, une méthode inspirée de l'algorithme de Gale-Shapley pour approximer les complexes microARN ::MRE formés à l'équilibre stœchiométrique. Elle s'est montrée plus précise et sensitive que les logiciels de prédiction de cibles largement répandus qui ne prennent pas en compte les abondances relatives d'ARN et de microARN.

Néanmoins, aucune méthode existante n'offre une réponse satisfaisante à la nature quantitative du microtargetome, donnant au mieux quelles interactions sont plus probables d'être fonctionnel sans pour autant s'avancer sur sa concentration. Ici, nous abordons cette question en proposant une refonte majeure de la méthode miRBooking qui réconcilie les propriétés biochimiques des interactions individuelles et l'idée d'une stœchiométrique globale dans un cadre dérivé de la cinématique de Michaelis-Menten. En particulier, notre approche permet un tout nouveau jeu de paradigmes basés sur l'analyse d'équilibre et la simulation numé-

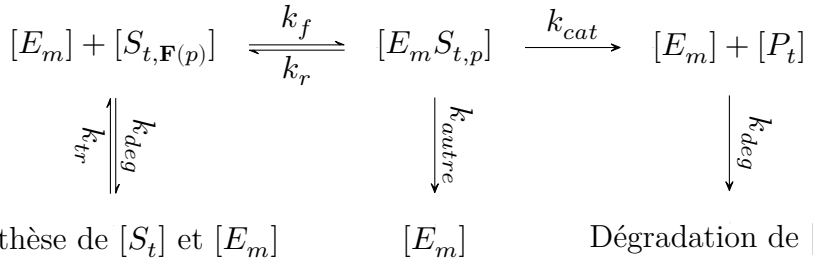


FIG. 2.1 : miRBooking 2.0 modélise le microtargetome par un ensemble de réactions enzymatiques impliquant des complexes $[E_m]$, des sites cibles $[S_{t,\mathbf{F}(p)}]$, des duplex formés $[E_m S_{t,p}]$ et des produits de dégradation $[P_t]$.

rique en temps réel afin d'inférer la nature du microtargetome et étudier sa dynamique. Notre contribution principale réside dans l'approche prise pour appliquer cette méthode à un système de dizaines de millions d'équations couplées que constitue le microtargetome humain.

2.1 Modèle biochimique

L'état du microtargetome $\mathcal{S}$ est caractérisé par les concentrations de quatre types d'espèces : les complexes RISC libres chacun chargé d'un microARN guide spécifique E_m, les ARN substrats ciblés S_t, les duplex RISC::MRE formés à des endroits sur ces substrats $[E_m S_{t,p}]$ et les produits de dégradation correspondant aux substrats $[P_t]$. Ces quantités sont toutes interconnectées par des réactions aux constantes spécifiques décrites dans la figure 2.1 et le tableau 2.1.

2.2 Modèle thermodynamique

Notre modèle thermodynamique répartit l'énergie libre du duplex en un ensemble de contributions additives provenant de l'amorce, des liaisons supplémentaires, l'accessibilité de l'ARN cible et un coût entropique assumé par le complexe RISC.

$$\Delta G = \Delta G_{\text{amorce}} + \Delta G_{\text{supplémentaire}} + \Delta G_{\text{accessibilité}} + \Delta G_{\text{AGO2}} \tag{2.1}$$

L'énergie contributoire de la région de l'amorce ΔG_{amorce} est estimée avec `RNAcofold` de ViennaRNA 2.4.11 [32] en hybridant chaque paire d'heptamères de la forme $t_8...t_2\&g_2...g_8$ sous contrainte d'une 4-mer canonique de g_2 à g_8 (voir figure 2.2). Nous utilisons l'argument `-p` afin d'extraire l'énergie de liaison telle que donnée par l'équation 2.2 au lieu de l'énergie

Constante de taux	Valeur de référence ou intervalle typique	Description
$k_{tr[E]}$	Exprimé en $\mathrm{pM\,s^{-1}}$	Biogenèse d'un complexe RISC
$k_{tr[S]}$	Exprimé en $\mathrm{pM\,s^{-1}}$	Transcription d'un ARN cible substrat
k_f	$1 \times 10^{-5}\,\mathrm{pM^{-1}\,s^{-1}}$ à $2.4 \times 10^{-4}\,\mathrm{pM^{-1}\,s^{-1}}$	Arrivée d'un complexe RISC guidé par m à la position p sur la cible t
k_r	$3.04 \times 10^{-4}\,\mathrm{s^{-1}}$ à $6.43\,\mathrm{s^{-1}}$	Dissociation d'un duplex RISC ::MRE
k_{cat}	$0\,\mathrm{s^{-1}}$ à $7.3 \times 10^{-3}\,\mathrm{s^{-1}}$	Conversion du substrat en produit
k_{autre}	Exprimé en $\mathrm{s^{-1}}$	Récupération de l'enzyme dû à l'activité sur la fraction partagée du substrat
$k_{deg[E]}$	$1.618 \times 10^{-6}\,\mathrm{s^{-1}}$	Dégradation d'un complexe RISC libre
$k_{deg[S]}$	$1.9254 \times 10^{-5}\,\mathrm{s^{-1}}$	Dégradation d'un ARN cible
$k_{deg[P]}$	$9.627 \times 10^{-5}\,\mathrm{s^{-1}}$	Dégradation d'un ARN cible coupé

TAB. 2.1 : Constantes de taux et valeurs typiques utilisées par miRBooking 2.0 pour modéliser les différentes parties du système biophysique (voir figure 2.1).

minimale afin de prendre en compte la contribution de toutes les structures secondaires. Les paires résultantes d'heptamères sont ensuite encodées en entier en base 4 et stockées efficacement à l'aide du codage Yale .

$$\Delta G_{\text{liaison}} = \frac{1}{Z} \sum_{\omega \in \Omega} e^{\frac{-\Delta G_\omega}{RT}} \Delta G_\omega \tag{2.2}$$

Les sites présentant une structure de *bulge* avec une guanine [33] sont pris en compte en ajoutant une contribution de $1.2\,\mathrm{kcal\,mol^{-1}}$ aux cibles présentant un `G` en position t_5 et en dérivant l'énergie de liaison de l'ensemble de structures résultant.

Nous estimons l'accessibilité de l'ARN $\Delta G_{accessibilit}$ en calculant le coût énergétique d'ouvrir 17 nucléotides – 9 en amont et 7 en aval de la position t_8 – à l'aide de `RNAup` [34] et GNU parallèle [35]. KERTESZ et al. [36] ont effectué un calcul similaire avec des fenêtres de 52 nucléotides, mais nous avons préféré nous en tenir au modèle d'empreinte du complexe.

La protéine Argonaute joue un rôle essentiel pour réduire le coût entropique associé à l'hybridation ARN-ARN en présentant le microARN guide dans une conformation favorable à

l'appariement [20]. Par conséquent, nous incluons une contribution de $-6.03\,\mathrm{kcal\,mol^{-1}}$ obtenue par maximum de vraisemblance. Ces valeurs sont consistantes avec les $-5.69\,\mathrm{kcal\,mol^{-1}}$ et $-5.47\,\mathrm{kcal\,mol^{-1}}$ estimées respectivement depuis Wee et al. [9] et Salomon et al. [10] pour des configurations où seul l'amorce est liée au substrat.

Les nucléotides supplémentaires sont considérés par deux modèles expérimentaux : les 3' supplémentaires $g_{13}...g_{16}$ de Wee et al. [9] et un modèle mécaniste proposé par Yan et al. [19] pour la queue du microARN.

Dans le premier modèle, nous considérons l'énergie de liaison des nucléotides $g_{13}...g_{16}$ en les calculant de la même manière que nous le faisons pour les amorces.

Pour le modèle de Yan et al. [19], l'automate à états finis qu'ils proposent se traduit sous la forme d'un modèle d'énergie séquentiel qui prend en compte tous les nucléotides du guide. Nous construisons un ensemble canonique (voir équation 2.3) comprenant tous les appariements qui satisfont les contraintes en calculons son énergie de liaison.

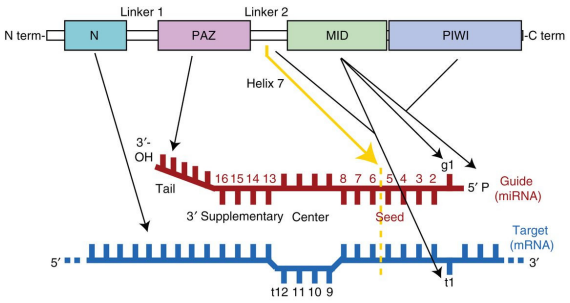


Fig. 2.2 : Correspondance entre les nucléotides du guide ($g_1 g_2 ...$) et de l'ARN cible ($t_1 t_2 ...$) ainsi que les domaines de la protéine Argonaute (N, PAZ, MID et PIWI) impliqués dans la formation du complexe RISC::MRE. Figure adaptée de Duchaine et Fabian [12].

$$\Delta G_{\text{supplémentaire}} = \Delta G_{\text{queue}} = \begin{cases} 0 \\ \Delta G_B \\ \Delta G_B + \Delta G_C \\ \Delta G_B + \Delta G_C + \Delta G_A \\ \Delta G_B + \Delta G_C + \Delta G_A + \Delta G_D \end{cases} \tag{2.3}$$

Les constantes de dissociations K_d et d'efficacités enzymatiques K_m sont dérivées du modèle d'énergie pour chaque duplex (voir équation 2.4 et 2.5) et exprimées en picomolaire.

$$K_d = 1 \times 10^{12} e^{\frac{\Delta G}{RT}} \tag{2.4}$$

$$K_m = K_d + \frac{k_{cat} + k_{autre}}{k_f} \tag{2.5}$$

Nous obtenons $k_f = 6.45 \times 10^{-5}\,\mathrm{pM^{-1}\,s^{-1}}$ comme constante de taux d'arrivée maximal, ce

qui est relativement proche du $2.4 \times 10^{-4}\,\mathrm{pM}^{-1}\,\mathrm{s}^{-1}$ obtenu expérimentalement par Salomon et al. [10]. Pour prendre en compte la variation du taux d'arrivée par des mésappariements dans la région de l'amorce [10, 26], nous multiplions des pénalités de 0.42 dans $g_2 ... g_5$ et 0.91 dans $g_6 ... g_8$ pour chaque cas observé. Par la suite, déterminer k_r est simple puisqu'il vaut $\frac{K_d}{k_f}$.

Puisque le clivage requiert que les nucléotides autour de $g_{10}g_{11}$ soient appariés, nous modulons un taux catalytique $k_{cat} = 2.62\,\mathrm{s}^{-1}$ par la probabilité que la région correspondante soit appariée. Dans le cas du modèle de Yan et al. [19], on considère la fraction des structures qui ont leur boîte A liée (quatrième et cinquième cas de l'équation 2.3).

Les paramètres libres : contribution entropique du complexe RISC, taux d'arrivée de base, taux catalytique de base et pénalités de mésappariements ont été obtenus par maximum de vraisemblance sur un jeu de données de mesures expérimentales [9, 10, 25, 26, 21, 37] et sont résumés dans la figure 2.3.

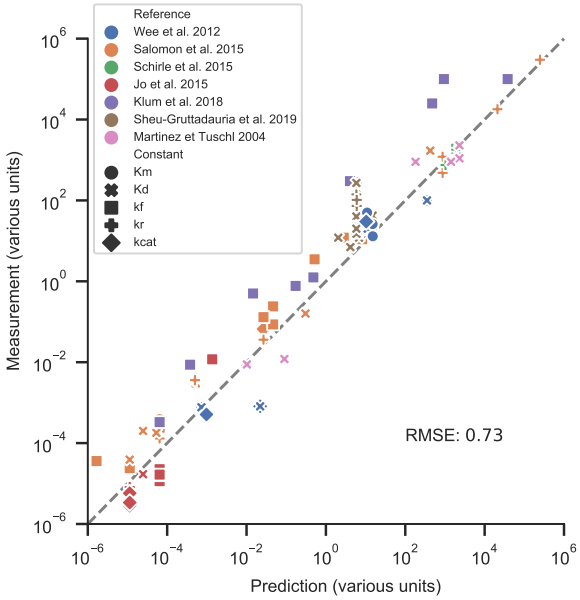


Fig. 2.3 : Taux et constantes d'équilibre dérivées du modèle thermodynamique proposé par le modèle (abscisse) comparé aux mesures expérimentales (ordonnée) comprenant des mesures de K_d, K_m, k_r, k_f and k_{cat}. La ligne grise pointillée indique un modèle idéal.

2.3 Concentration d'un site sur une cible

Une propriété importante de miRBooking 2.0 est sa définition de concentration pour un site sur une cible prenant en compte les autres complexes formés dans le voisinage.

L'empreinte d'une position p est définie par l'ensemble des positions entourant p qui est occupée par un complexe RISC . Par convention, nous utilisons la position t_8 sur la cible correspondant au dernier nucléotide du guide g_8 comme référence. En nous appuyant sur Sætrom et al. [38], nous déterminons que 17 nucléotides (c.-à-d. 9 en amont et 7 en aval) est suffisante pour faire l'hypothèse d'indépendance entre deux complexes.

$$\mathbf{F}(p_i) = \{p_{i-9}, ..., p_i, ..., p_{i+7}\} \tag{2.6}$$

La concentration d'une position est donnée par sa concentration libre de chevauchements des complexes RISC proximaux.

$$[S_{t,p}] = [S_t] - \sum_m \sum_{p' | p \in \mathbf{F}(p')} [E_m S_{t,p'}] \tag{2.7}$$

On s'intéresse par contre à la concentration du segment qu'occupera ce complexe autour de p via son empreinte $\mathbf{F}(p)$, ce qui nécessite de calculer la distribution jointe sur l'ensemble des positions étant libres.

$$\Pr[\mathbf{F}(p) \text{ est libre}] \tag{2.8}$$

Heureusement, ce calcul peut être effectué très efficacement en factorisant la distribution sur les positions.

$$\prod_{p_i \in p_1 \cdots p_n} \Pr[p_i \text{ est libre} \mid p_1 ... p_{i-1} \text{ sont libres}] \tag{2.9}$$

La première position p_1 est directement estimée par $1 - \sum_{\{p | p_1 \in \mathbf{F}(p)\}} \frac{[S_{t,p}]}{[S_t]}$, la fraction du substrat pour laquelle la position est libre de chevauchements. Qu'en est-il des autres cas ? La formule est très similaire puisqu'on doit calculer une fraction libre, mais conditionnée de telle sorte que toutes les positions précédentes soient libres elles aussi.

$$\prod_{i=1}^{n} \left[1 - \underbrace{\{p_j | p_i \in \mathbf{F}(p_j)\}}_{\text{positions chevauchant } p_i} - \underbrace{\cup_{k<i} \{p_k | p_i \in \mathbf{F}(p_k)\}}_{\text{positions chevauchant une position précédente à } p_i} \frac{[S_{t,p_j}]}{[S_t]} \right]$$

La concentration disponible du segment est obtenue en multipliant la fraction libre par la concentration du substrat.

$$[S_{t,\mathbf{F}(p)}] = \Pr[\mathbf{F}(p) \text{ est libre}] \times [S_t] \tag{2.10}$$

De manière intéressante, cette astuce peut être appliquée pour modéliser n'importe quelle protéine qui se lie à l'ARN pour autant qu'elle occupe un ensemble bien défini de positions.

2.4 Nombre d'occupants sur une cible

Même s'il est possible d'étendre notre définition de concentration d'un site au substrat entier, cette approche ne fonctionne pas lorsqu'on cherche à calculer la fonction de masse de cette distribution. Déterminer la probabilité d'avoir k positions occupées requiert $\binom{k}{n}$ factorisations, n étant le nombre de sites qui croît proportionnellement à la longueur du transcrit.

Introduire l'hypothèse d'indépendance nous permet d'utiliser une distribution Poisson-Binomiale pour modéliser le nombre d'occupants sur une cible. Cette distribution modélise le nombre de succès parmi n expériences de Bernoulli non nécessairement identiques aux probabilités $\mathbf{p} = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n\}$.

$$\Pr[n = k] = \sum_{A \in [\Omega]^k} \prod_{i \in A} \mathbf{p}_i \prod_{j \in A^{\complement}} (1 - \mathbf{p}_j) \tag{2.11}$$

Un paramètre $\mathbf{p}_i$ est estimé par le nombre total de microARN qui occupent exactement l'i-ème site sur la cible sans prendre en compte les chevauchements des sites proximaux.

$$\mathbf{p}_i = p_{t,i} = \frac{\sum_m [E_m S_{t,i}]}{[S_t]} \tag{2.12}$$

La complexité d'énumérer tous les sous-ensembles de taille k d'Ω est prohibitive et une forme close existe pour calculer la fonction de masse de cette distribution en $O(n^2)$ par une transformée de Fourier discrète [39].

Les cas particulièrement importants de cette distribution sont $\Pr[k = 0]$, la probabilité d'observer aucun microARN et $\Pr[k > 0] = 1 - \Pr[k = 0]$, la fraction occupée du substrat. Le nombre espéré de complexes RISC par unité de substrat est donné par l'espérance $\sum_{m,p} \frac{[E_m S_{t,p}]}{[S_t]}$ qui correspond naturellement au nombre moyen de microARN par cible.

2.5 Activité enzymatique sur une séquence

Le dernier terme du modèle k_{autre} permet de prendre en compte l'activité catalytique des autres complexes RISC formés tout au long de l'ARN cible. Lorsqu'un événement de clivage se produit, nous présumons que les fragments décoiffés et désanlyés résultants sont rapidement dégradés par les endonucléases, libérant du fait même les autres complexes formés en amont et en aval.

Évidemment, si deux complexes distincts sont à moins d'une distance d'empreinte l'un de l'autre, ils ne peuvent pas occuper la même unité du substrat, mais s'ils sont suffisamment éloignés, ils partageront une fraction commune du substrat.

De la perspective d'un duplex, le nombre espéré de co-occupants par unité de substrat occupée est obtenu en appliquant l'espérance de la distribution Poisson-Binomial. Chaque co-occupant contribue un k_{cat} spécifique à l'activité catalytique totale k_{autre} donné par l'équation 2.13.

$$k_{autre} = \sum_{m',p' | \mathbf{F}(p') \cap \mathbf{F}(p) = \emptyset} \frac{k_{cat}[E_{m'}S_{t,p'}]}{[S_t]} \tag{2.13}$$

2.6 Équations différentielles partielles

Nous présentons ici une simplification du modèle proposé pour une seule réaction. On omet spécifiquement les sommes sur les enzymes, substrats et produits qui sont impliqués dans plusieurs réactions.

$$\begin{aligned}
\frac{\partial[E_m]}{\partial t} &= k_{tr} - k_f[E_m][S_{t,\mathbf{F}(p)}] + (k_r + k_{cat} + k_{autre})[E_m S_{t,p}] - k_{deg}[E_m] \\
\frac{\partial[S_t]}{\partial t} &= k_{tr} - k_{cat}[S_{t,p}] - k_{deg}[S_t] \\
\frac{\partial[E_m S_{t,p}]}{\partial t} &= k_f[E_m][S_{t,\mathbf{F}(p)}] - (k_r + k_{cat} + k_{autre})[E_m S_{t,p}] \\
\frac{\partial[P_t]}{\partial t} &= k_{cat}[E_m S_{t,p}] - k_{deg}[P_t]
\end{aligned}$$

2.7 Recherche de l'état d'équilibre

L'état d'équilibre $\mathcal{S}^0$ et caractérisé par une assignation des variables telle que l'ensemble des équations différentielles sont simultanément nulles. Dans le cas d'un système reposant

sur le principe d'action de masse tel que nous avons ici, cet état est unique et existe [40] et le trouver constitue un problème bien posé.

Pour obtenir $\mathcal{S}^0$, nous supposons un taux égal de transcription et de dégradation du substrat ainsi qu'une concentration stationnaire de produit en choisissant $_{k_{tr}}$ et $[P]$ de telle sorte que $\frac{\partial[E]}{\partial t}$, $\frac{\partial[S]}{\partial t}$ et $\frac{\partial[P]}{\partial t}$ soient toujours nuls.

Sous l'hypothèse d'un état d'équilibre, nous pouvons réduire substantiellement le coût computationnel en réduisant le système à résoudre aux concentrations des complexes $[ES]$ et en inférant les autres variables à partir des conditions initiales via $[E_m] = [E_m]_0 - \sum[E_m S_{t,p}]$ et $[S_t] = [S_t]_0$

$$[ES]^{(t+1)} = [ES]^{(t)} - J^{-1}\frac{\partial[ES]}{\partial t} \tag{2.14}$$

La racine du système est obtenue par la méthode de Newton-Raphson (voir équation 2.14). Cette méthode requiert l'évaluation d'une matrice jacobienne qui lie chaque dérivée partielle $\frac{\partial[ES]}{\partial t}$ à chaque variable $[ES]$ du système.

$$-J([ES]^{(t+1)} - [ES]^{(t)}) = \frac{\partial[ES]}{\partial t} \tag{2.15}$$

Pour chaque équation modélisée, nous obtenons l'expression analytique de chaque dérivée partielle requise pour déterminer J. Inverser de grandes matrices n'est pas particulièrement efficace, alors l'itération est réorganisée pour obtenir un système d'équation linéaire (voir équation 2.15) qui est résolue par la suite par une décomposition LU et une substitution arrière.

Nous montrons, via nos dérivées analytiques, que le Jacobien du système est largement composée de zéros (p. ex. >95% éparse en pratique) et les méthodes permettant la résolution de ce type de systèmes sont très efficaces [41, 42, 43].

La matrice exhibe aussi deux importantes propriétés : elle est structurellement symétrique et définie négative, ce qui peut être bénéfique à certains solveurs linéaires épars. Remarquablement, cette éparsité tient même en prenant en compte les complexes RISC se chevauchant lorsqu'on dérive l'équation 2.10.

Même si le terme k_{autre} induit des valeurs non nulles pour des complexes indépendants occupant le même transcrit, ignorer ces contributions dans le Jacobien n'affecte pas significativement la convergence pour autant qu'elles soient prises en compte dans la diagonale principale.

2.8 Intégration numérique

L'intégration numérique est effectuée à l'aide de la méthode de Dormand-Prince [44] qui possède une erreur de troncation locale de l'ordre de $O(h^5)$.

2.9 Benchmarks HITS-CLIP

Nous avons comparé notre méthode à celles déjà établies à expliquer la taille des pics observés expérimentalement par HITS-CLIP [4].

En premier lieu, nous avons traité les lectures de séquençage avec `cutadapt` [45] et `umi_tools` [46] afin de retirer les adaptateurs et extraire les codes-barres. Ensuite, les lectures ont été alignées sur la version 19 du génome humain annoté par GENCODE v19 avec STAR [47] et les paramètres par défaut utilisés par CLIPSeqTools [48]. Les alignements résultats ont été dédupliqués et les pics ont été détectés avec Piranha [49] dans des fenêtres de 200 nucléotides. À l'exception du réplicat B de l'anticorps 7G1, tous les réplicats ont atteint 24% d'alignements uniques de 29 nucléotides en moyenne. Le *workflow* exact est disponible dans le matériel supplémentaire en format WDL [50]. Cette analyse a produit 25 352 pics qui corrèlent relativement bien ($\rho = 0.7$).

Pour utiliser miRBooking, il est nécessaire d'avoir les profils d'expression des ARN longs et petits d'un modèle cellulaire représentatif. Nous avons donc sélectionné l'épigénome de référence d'HeLa S3 du projet ENCODE (ENCSR068MRQ) pour reproduire les conditions de contrôle et de surexpression de miR-124 *in silico*. Les quantifications de transcrits pour GENCODE v19 sont fournies directement par ENCODE en FPKM. Les quantifications de microARN mature ont été obtenues en quantifiant les alignements avec HTSeq [51] sur miRBase v21. La conversion in picomolaire des abondances est basée sur une estimation du nombre d'ARN codants (7e5 de la spécification de Thermofisher), de microARN (1 4e5 depuis Wang et al. [52]) et du volume cytoplasmique (940 µm^3 depuis Fujioka et al. [53]) pour une cellule HeLa.

Le ratio picomolaire-par-FPKM est obtenu en divisant le nombre total de FPKM attribué à une classe d'ARN à sa concentration totale attendue (voir équation 2.16). Idéalement, il faudrait mesurer différents transcrits connus pour être localisé dans le cytoplasme à l'aide d'une méthode quantitative comme RT-qPCR et faire une régression log-linéaire pour trouver les vrais ratios.

$$c_{pM} = \frac{1 \times 10^{12} \times N_{molecules}}{N_A \times 3 \times 10^{-15} \times V_{cytoplasm}} \tag{2.16}$$

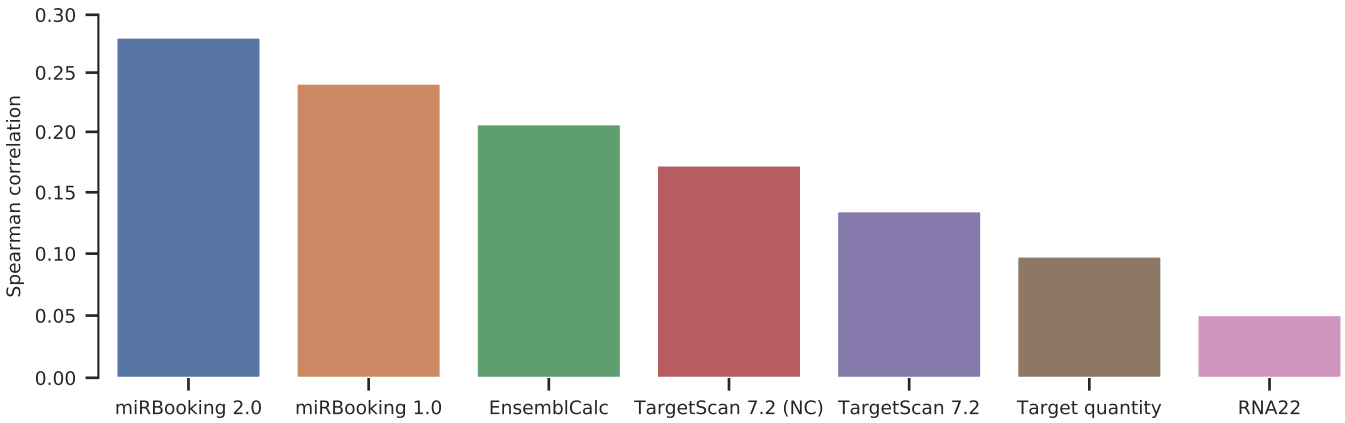


FIG. 2.4 : Corrélation de rang (Spearman) entre les pics mesurés expérimentalement et les scores prédits par les méthodes considérées. L'ordonnée utilise une transformée de Fisher.

Pour TargetScan 7.2 [5], nous avons sélectionné les scores context++ pour les sites conservés et non conservés dans les régions 3'UTR des transcrits codants et imputé chaque pic par le meilleur score lorsque plusieurs recoupements existent.

Pour RNA22 [54], nous avons sélectionné les meilleures prédictions en filtrant celles avec une signifiance $\alpha \leq 0.01$ et imputé les pics par le meilleur score de chaque pic.

$$[ES] = \frac{Z \pm \sqrt{(Z^2 - 4[E]_0[S]_0}}{2} \qquad Z = [E]_0 + [S]_0 + K_m \tag{2.17}$$

Afin de comparer équitablement nos prédictions avec celles d'EnsemblCalc [30], nous utilisons le même modèle d'énergie et les quantifications d'espèces que celles utilisées pour miRBooking 2.0. Les concentrations de complexes sont obtenues par la solution analytique (voir l'équation 2.17).

2.10 L'équilibre global explique les abondances des pics de HITS-CLIP

Nous avons tout d'abord évalué la capacité des méthodes à prédire la taille des pics d'AGO2 d'un jeu de données HITS-CLIP[4, 31] (voir figure 2.4). Notre méthode explique le mieux les données avec une corrélation de rang de $\rho = 0.28$ (P-value : 6.9×10^{-30}) tel que détaillé dans la figure 2.5.

Remarquablement, la solution obtenue par l'équilibre global considéré par notre modèle est beaucoup plus fidèle que celle donnée par EnsemblCalc même si les deux méthodes

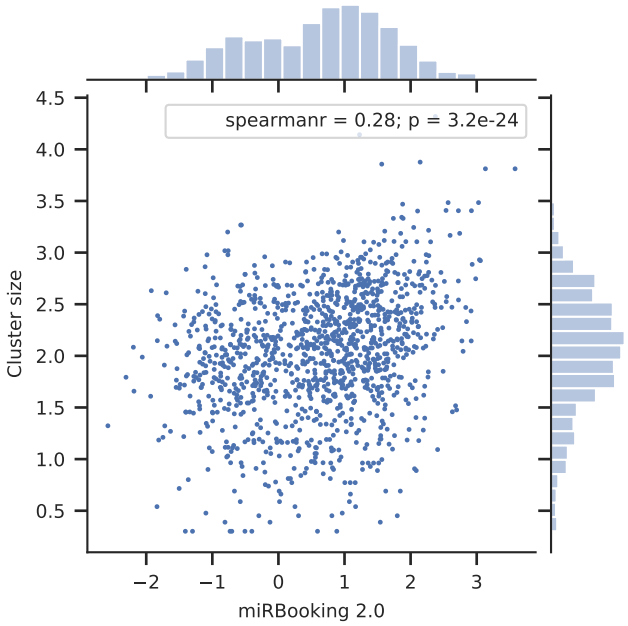


FIG. 2.5 : Nuage de points comparant les prédictions du modèle miRBooking (en abscisse) avec les valeurs obtenues par HITS-CLIP (en ordonnée).

utilisent le même modèle thermodynamique et les mêmes concentrations.

La solution obtenue par EnsemblCalc, ou n'importe quel sous-ensemble de l'interactome se comporte comme une borne supérieure à la concentration du duplex (voir figure 2.6. Cette inégalité est très utile, car elle nous permet de filtrer les interactions insignifiantes d'emblée afin de réduire le coût computationnel.

2.11 Les prédictions sont robustes au bruit technique et biologique

Pour évaluer la robustesse de notre modèle à la volatilité des estimations de concentrations, nous avons utilisé un jeu de données de co-séquençage de cellules individuelles de WANG et al. [7] qui ont réussi à séquencer les petits et longs ARN de 19 cellules individuelles. À partir de ces données, nous avons modélisé 19 microtargetomes : les prédictions de concentration de duplexes ont démontré une grande reproductibilité ($R^2 = 0.92$) lorsque comparées par paires (voir la figure 2.7).

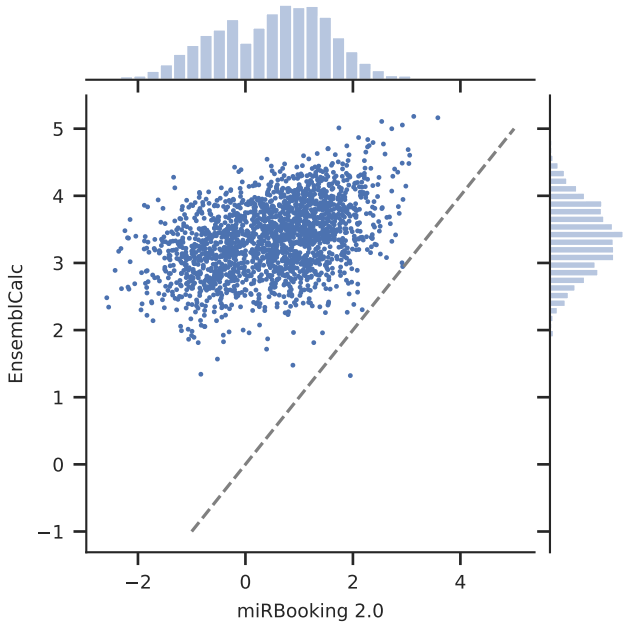


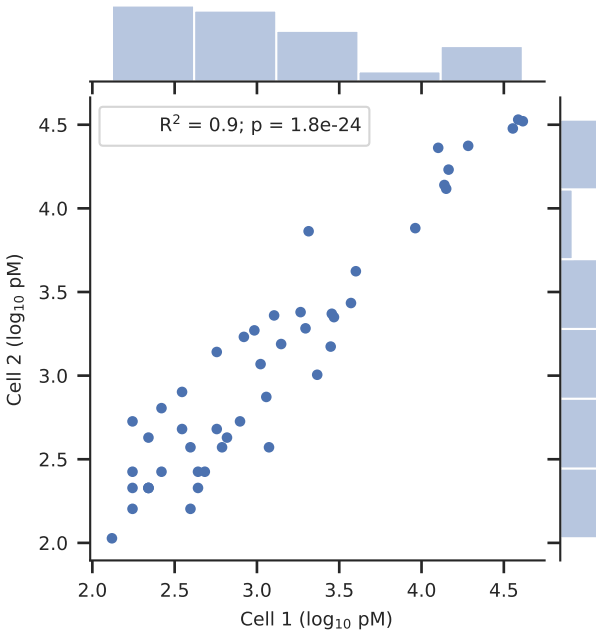
Fig. 2.6 : L'équilibre obtenu par le modèle unidimensionnel surestime systématiquement la concentration à l'équilibre global.

2.12 Calculer efficacement l'équilibre du microtargetome

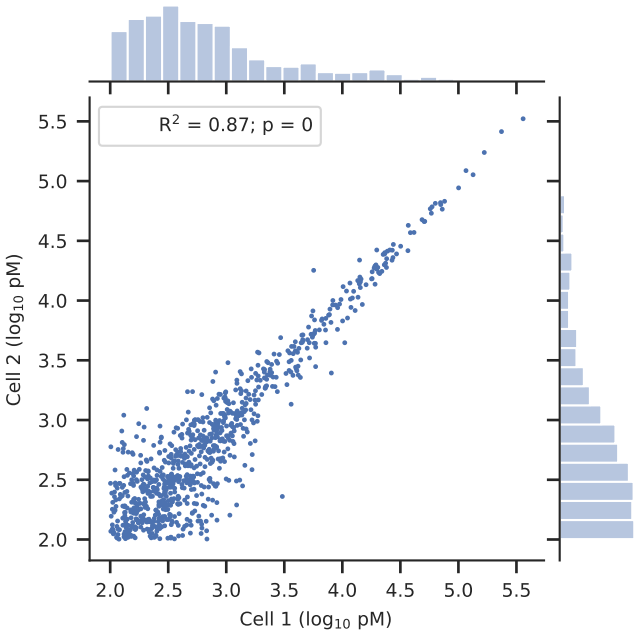
Nous réussissons à appliquer notre méthode à l'échelle du microtargetome humain en exploitant l'éparsité de la matrice jacobienne du système (voir figure 2.8), dont les valeurs non nulles émergent uniquement lorsqu'une paire de duplex possède un microARN en commun ou cible le même site. Surprenamment, la dérivée s'annule lorsqu'une paire de duplex occupe des sites qui se chevauchent sans pour autant occuper la même position de référence. En pratique, nous atteignons $\geq 95\%$ d'éparsité, permettant la résolution de systèmes dont le Jacobien comporte plusieurs milliards d'entrées.

Le temps et la mémoire requise pour résoudre le système croient selon le nombre d'équations à résoudre et nous projetons résoudre des systèmes beaucoup plus grands en distribuant la matrice jacobienne sur plusieurs nœuds via MPI pour passer la limite de 1 TB de mémoire. Ces fonctionnalités sont déjà supportées par le solveur épars Intel MKL Cluster.

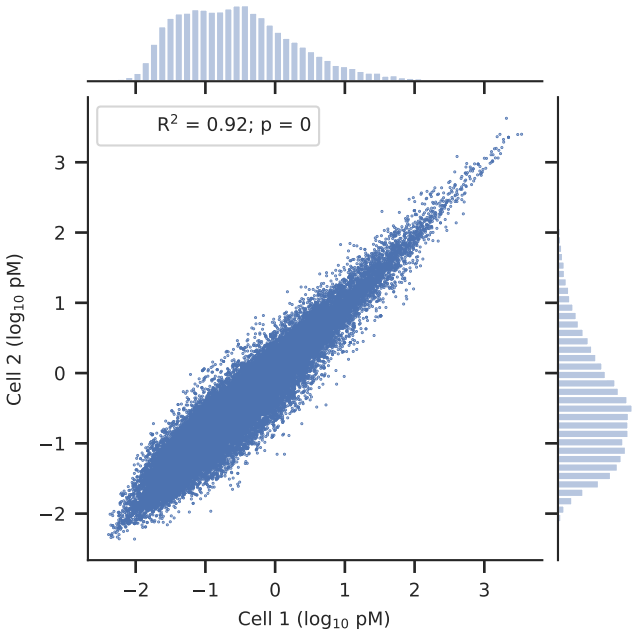
2.13 miRBooking 2.0 modélise les effets de seuil d'expression des gènes



(a) Concentrations des microARN $[E]_0$



(b) Concentrations des transcrits cibles $[S]_0$



(c) Concentrations des complexes prédits $[ES]$

FIG. 2.7 : La dispersion des réplicats de microARN et transcrits cibles (2.7a, 2.7b) avec celle des prédictions résultantes (2.7c) indique que le modèle est robuste à la variation biologique et technique.

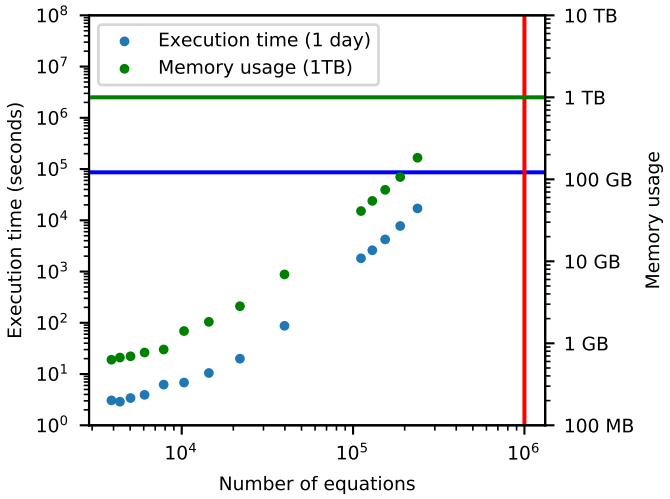


FIG. 2.9 : Temps d'exécution et mémoire utilisé pour résoudre l'état d'équilibre de sous-ensembles du microtargetome de différentes tailles en fonction du nombre d'équations. Les lignes bleu et verte représentent des limites pratiques de ressources. Les mesures ont été faites avec le solveur Intel MKL PARDISO sur 16 processeurs Intel Xeon Gold 6130 capables d'exécuter jusqu'à 64 fils d'exécutions.

Nous reproduisons les résultats de MUKHERJI et al. [55] sur les seuils d'expression que causent les microARN. En bref, leurs résultats indiquent que les microARN créent des effets de seuils en séparant la traduction de l'ARNm en deux régimes. Sous le seuil, l'ARN est fortement réprimé et peu exprimé sous forme de protéine et, à partir d'un certain niveau, échappe à la régulation pour être asymptotiquement entièrement exprimé. En variant le nombre de sites de liaison de miR-20a sur un rapporteur mCherry et la concentration du microARN, les auteurs montrent que le régime sous le seuil et sa position peuvent être modifiés.

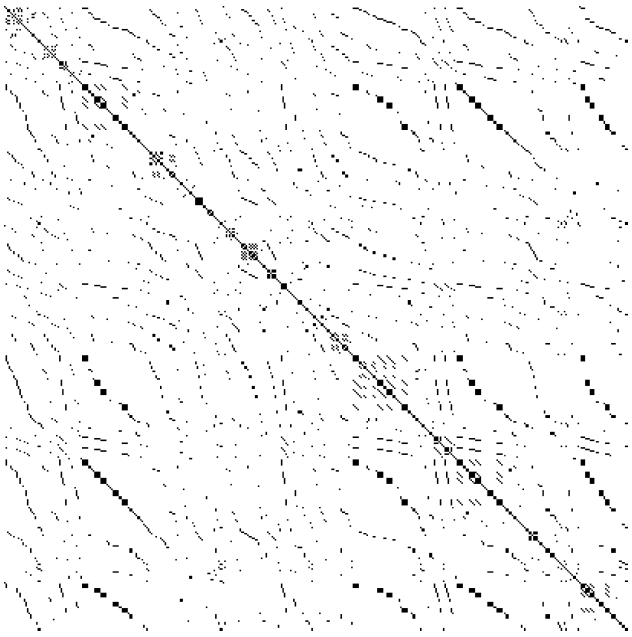


FIG. 2.8 : Le Jacobien du système est extrêmement éparse, permettant la résolution de très grands systèmes en utilisant la méthode de Newton-Raphson pour en trouver le zéro.

$$[S_t]_{\text{traduit}} = [S_t] \times \sum_{k=0}^{n} \Pr[k \text{ miRNA on } t] e^{-\lambda k} \tag{2.18}$$

Puisque notre modèle ne prend pas en compte l'effet sur la protéine, nous proposons

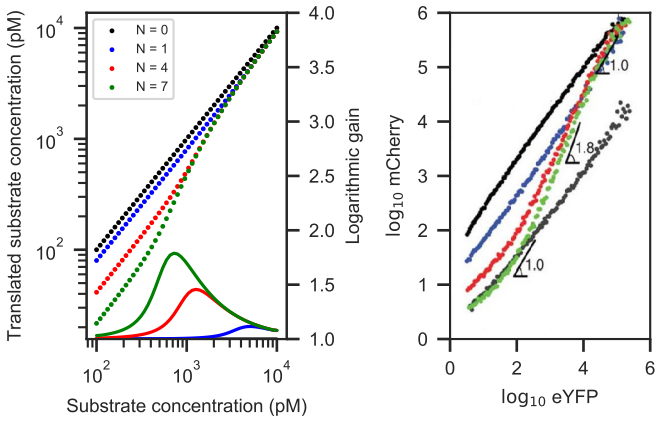


FIG. 2.10 : Les réponses prédites (à gauche) en titrant différentes concentrations d'un reporteur contenant N sites de liaisons de miR-20a dans sa région 3'UTR reproduisent très fidèlement les résultats obtenus expérimentalement par MUKHERJI et al. [55] (à droite).

un modèle log-linéaire (voir équation 2.18) où le nombre de microARN liés affecte exponentiellement la fraction de l'ARN traduit. Il suffit de faire l'expansion de la distribution Poisson-Binomiale pour connaître le poids de chaque cas discret de nombre d'occupants. Le paramètre $\lambda = 0.27$ est ajusté de sorte que le gain logarithmique à la position du seuil corresponde à celui mesuré par MUKHERJI et al. [55] pour 7 sites de liaison sur le rapporteur ($N = 7$ dans la figure). Il est intéressant de noter que ce paramètre est indépendant des effets d'échelle comme un changement des unités de concentration puisque cela correspondrait à une translation du graphique log-log et n'affecterait pas la pente observée.

Ce modèle simple permet d'expliquer remarquablement les courbes observées expérimentalement (voir figure 2.10) en reproduisant autant les régimes sous le seuil que la position de ce dernier. Ce résultat soutient l'idée que le microtargetome est explicable par un modèle d'équilibre stœchiométrique global tel que celui que nous proposons.

2.14 miRBooking simule le microtargetome en temps réel

Nous complémentons notre travail avec CHI et al. [4] en simulant la surexpression de miR-124. En premier lieu, nous atteignons l'état d'équilibre afin d'obtenir les conditions initiales nécessaires pour l'intégration numérique. Nous introduisons ensuite 10 nM de complexes RISC chargés avec miR-124 et nous simulons la réponse sur une période d'une heure.

La surexpression augmente l'activité endonucléique à travers le microtargetome qui répond par une diminution des concentrations des substrats jusqu'à ce que leurs taux cataly-

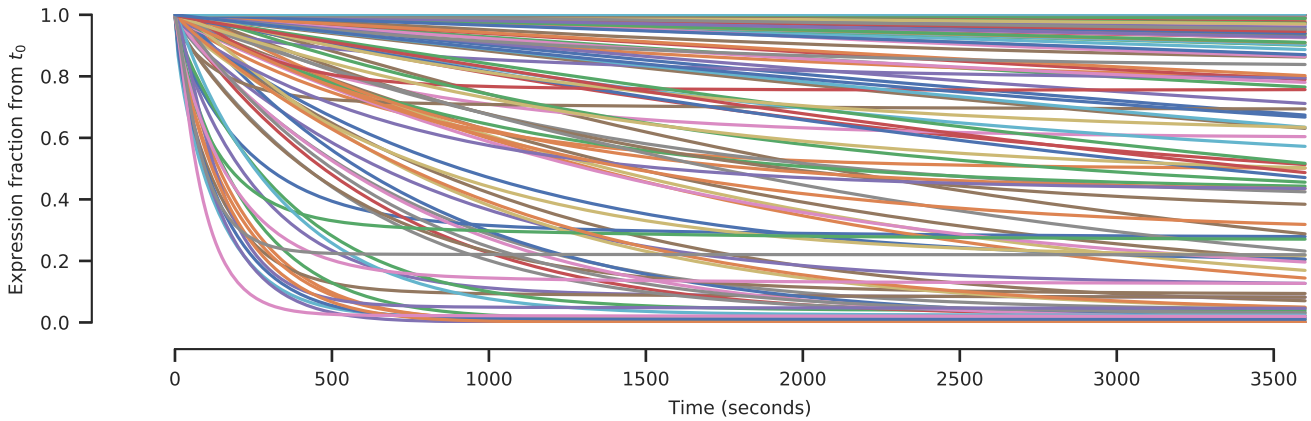


FIG. 2.11 : La réponse des gènes cibles est simulée en temps réel à l'aide d'une méthode d'intégration numérique. À l'équilibre, la fraction de la concentration restante varie de 2,36% pour ALG5 jusqu'à 95,15% pour DUT.

tiques atteignent leur taux de transcriptions fixé depuis l'état d'équilibre. La concentration à laquelle ces substrats terminent leurs parcours est unique et varie en fonction du nombre de sites de miR-124 qu'ils possèdent et des propriétés stœchiométriques du système.

2.15 Remarques et conclusions

miRBooking 2.0 encapsule d'importantes pièces du casse-tête que représente la régulation post-transcriptionnelle des gènes dans un modèle biochimique pratique qui peut mener à un large éventail d'applications. Plus important encore, nous montrons que l'état d'équilibre et la simulation de très grands systèmes est possible avec des ressources computationnelles modestes. Nous proposons également un modèle statistique pour le problème général de régulation de séquence lorsque le complexe qui s'y associe occupe un intervalle bien défini de positions.

Il reste fondamental de garder à l'esprit que les prédictions du modèle sont faites sous l'hypothèse d'un état d'équilibre, ce qui requiert des hypothèses qui ne reflètent pas nécessairement la réalité des cellules individuelles d'un échantillon biologique. Ce qui est observé via séquençage à haut débit n'est qu'un aperçu de l'état transitoire d'une population de cellules tandis que les prédictions de notre modèle portent sur l'état asymptotique satisfaisant ces observations.

De plus, dû au très faible nombre de jeux de données pour les autres protéines de la famille Argonaute, le modèle que nous proposons est spécifique à AGO2. En particulier, AGO1, AGO2 et AGO3 ne sont pas capables de catalyser leurs substrats et reposent plutôt sur d'autres éléments de la voie de dégradation des ARN pour accomplir ce travail.

Notre modèle assume aussi une distribution uniforme des complexes RISC et des ARN cibles à travers le cytoplasme, alors qu'on sait qu'ils sont colocalisés dans les P-bodies, des granules compactes d'ARN qui agissent comme centre de stockages et de dégradation pour les ARNm [56]. Le volume où les réactions se produisent pourrait être beaucoup plus petit, menant à des interactions beaucoup plus fortes que prévu.

2.16 Disponibilité des données et du code source

Tout le code source de miRBooking 2.0 incluant les scores précalculés des heptamères des amorces et des nucléotides supplémentaires est rendu disponible au https://major.iric.ca/mirbooking/ sous la licence de logiciel libre MIT.

miRBooking 2,0 supporte une variété de solveurs linéaires épars parallèles, distribués et adaptés pour le calcul sur GPGPU. Il a été testé avec les séquences de références de RefSeq, GenBank, GENCODE et miRBase. Des *bindings* Python sont disponibles pour rendre l'utilisation du modèle conviviale dans en environnement interactif.

Des prédictions précalculées sont disponibles en ligne pour un grand nombre de lignées cellulaires et tissus au https://major.iric.ca/~poirigui/mirbooking-scan/.

Chapitre 3

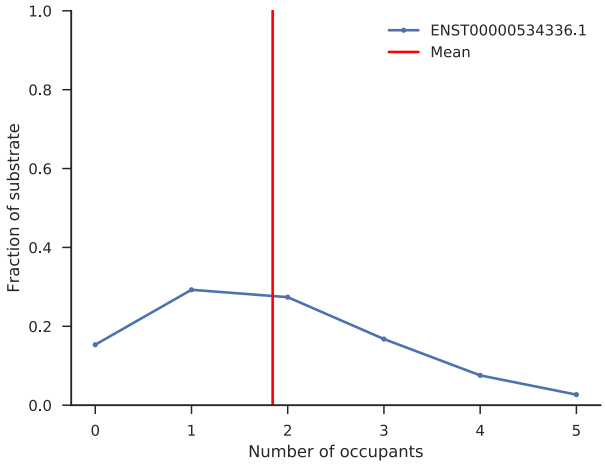
Propriétés mathématiques de miRBooking 2.0

Ce chapitre porte sur les propriétés mathématiques du système d'équations différentielles étudié et modélisé numériquement dans la méthode miRBooking 2.0.

3.1 Distribution Poisson-Binomiale

On dérive de notre modèle la probabilité d'observer k microARN sur une unité du substrat cible à l'aide de la distribution Poisson-Binomiale.

La distribution Poisson-Binomiale [57] établit la probabilité d'observer k succès parmi n épreuves de Bernoulli indépendantes aux probabilités de succès $\mathbf{p} = \{p_1, ..., p_n\}$. Elle généralise la distribution binomiale dans la mesure où elle ne contraint pas les épreuves à être identiquement distribuées.



Exemple de fonction de masse d'une distribution Poisson-Binomiale modélisant le nombre de microARN hybridés sur le variant MALAT-202 du gène non-codant MALAT1.

Malgré le fait que les événements d'hybridation ne sont pas nécessairement indépendants puisque les microARN compétitionnent pour les sites voisins, en considérant le cas très général de l'activité totale où la grande majorité des interactions sont indépendantes, on se permet d'en faire l'assomption.

Dans le chapitre précédent, on calcule spécifiquement le cas $\Pr[k = 0]$ par factorisation de la distribution jointe (voir équation 2.9) pour calculer la concentration de substrat disponible et on indique aussi que le calcule serait très laborieux pour les autres valeurs de k. Pour cette raison, il est très intéressant de se référer au modèle statistique Poisson-Binomiale pour les analyses et l'interprétation.

Pour appliquer ce modèle à notre problème, la première étape est de constituer les paramètres de notre distribution. On considère que le nombre d'événements indépendants correspond au nombre de positions occupées par différent microARN sur un segment donné d'ARN. Chaque position est caractérisée par une probabilité d'y observer un microARN que l'on obtient en prenant le rapport entre la concentration totale de microARN qui l'occupe et la concentration du substrat.

$$p_i = \Pr[\text{microARN sur } S_{t,i}] = \sum_m \frac{[E_m S_{t,i}]}{[S_t]} \tag{3.1}$$

Par la fonction de masse de cette distribution, on peut calculer la probabilité d'observer k microARN sur un ARN cible :

$$\Pr[k \text{ microARN sur } S_t] = \sum_{A \in \mathcal{P}_k(S_t)} \prod_{p \in A} p \prod_{p \in A^{\complement}} (1 - p) \tag{3.2}$$

où $\mathcal{P}_k(S_t)$ est l'ensemble des sous-ensembles de k sites de liaison sur le substrat S_t.

L'espérance et la variance du nombre de microARN sur une cible sont faciles à dériver puisqu'il s'agit d'un ensemble indépendant d'expériences de Bernoulli :

$$\mathbb{E}[\text{microARN sur } S_t] = \sum_{i=1}^{n} p_i \tag{3.3}$$

$$\mathbb{V}\mathrm{ar}[\text{microARN sur } S_t] = \sum_{i=1}^{n} p_i(1 - p_i) \tag{3.4}$$

Évaluation de la fonction de masse

Dans le cadre du projet miRBooking 2.0, deux implémentations de la fonction de masse de cette distribution ont été réalisées en utilisant la forme close par transformée de Fourier avec FFTW [39] et par programmation dynamique avec la récurrence [58]. Cette dernière comporte des problèmes de stabilité numérique pour des instances de plus de 20 événements.

Pour les analyses en Python, l'implémentation de J. Straka [59] a été réutilisée.

Quoique numériquement instable, la méthode par récurrence peut toutefois être considérablement accélérée en évaluant intelligemment la fonction de masse afin de réutiliser des calculs précédents.

En premier lieu, voici la forme de cette récurrence :

$$\Pr[k] = \begin{cases} \prod(1-p_i) & k = 0 \\ \frac{1}{k}\sum_{i=1}^{k}(-1)^{i-1}\Pr[k-i]T(i) & k > 0 \end{cases} \tag{3.5}$$

$$T(i) = \sum_{j=1}^{n}(\frac{p_j}{1-p_j})^i \tag{3.6}$$

Pour établir la complexité, considérons les faits suivants :

- $T(j)$ prend un temps $\Theta(n)$, peu importe l'argument fourni
- $\Pr[0]$ se fait en $\Theta(n)$

Évaluer $\Pr[k = n]$ nécessite $n!$ appels puisque chaque appel de $\Pr[k]$ demande à faire k évaluation récursive de $\Pr[k-1]$. Puisqu'on évalue $T(j)$ à chaque fois, la complexité finale est donc de $\Theta(nn!)$.

Or, on remarque que l'évaluation de $\Pr[k]$ requiert l'évaluation préalable des termes $\Pr[1]$ jusqu'à $\Pr[k-1]$. En évaluant en ordre et en réutilisant ces calculs, $\Pr[n]$ coûte toujours un seul appel de $T(j)$ et la somme coûte seulement $\Theta(n^2)$ appels récursifs. Au total, on obtient une complexité de $\Theta(n^3)$.

De plus, $T(j)$ peut être précalculé pour tous les j possibles en un temps $\Theta(n^2)$ et stocké dans un tableau réduisant ainsi la complexité à $\Theta(n^2)$.

Les implémentations en C des deux approches sont fournies en annexe.

3.2 Équations différentielles

Le modèle d'équations différentielles suivant permet de généraliser le modèle de Michaelis-Menten pour plusieurs enzymes et substrats.

$$\frac{\partial[E_m]}{\partial t} = \sum_t \sum_{p \in t} -k_f[E_m][S_{t,\mathbf{F}(p)}] + k_r[E_m S_{t,p}] + k_{cat}[E_m S_{t,p}] + \left(\sum_{m'} \sum_{p' | \mathbf{F}(p) \cap \mathbf{F}(p') = \emptyset} \frac{k_{cat}[E_{m'} S_{t,p'}]}{[S_t]} \right) [E_m S_{t,p}] \tag{3.7}$$

$$\frac{\partial[S_{t,p}]}{\partial t} = \sum_m \sum_{p' | \mathbf{F}(p) \cap \mathbf{F}(p') \neq \emptyset} -k_f[E_m][S_{t,p'}] + k_r[E_m S_{t,p'}] \tag{3.8}$$

$$\frac{\partial[E_m S_{t,p}]}{\partial t} = k_f[E_m][S_{t,p}] - k_r[E_m S_{t,\mathbf{F}(p)}] - k_{cat}[E_m S_{t,p}] - \left(\sum_{m'} \sum_{p' | \mathbf{F}(p) \cap \mathbf{F}(p') = \emptyset} \frac{k_{cat}[E_{m'} S_{t,p'}]}{[S_t]} \right) [E_m S_{t,p}] \tag{3.9}$$

$$\frac{\partial[P_t]}{\partial t} = \sum_m \sum_{p \in t} k_{cat}[E_m S_{t,p}] \tag{3.10}$$

Pour éviter les abus de notation, on considère que les constantes de taux k_f, k_r et k_{cat} correspondent aux termes multipliés.

Lorsqu'un transcrit est catalysé par un autre microARN m' (voir équation 3.7), la proportion des $[E_m]$ qui sont hybridé sur les mêmes copies à l'extérieur de son empreinte sont récupérés. En particulier, si la fenêtre de deux microARN possède une intersection, ils seront nécessairement assignés sur des copies différentes et n'influenceront pas leurs concentrations mutuelles.

Pour résoudre une instance spécifique, il faut prendre en compte les concentrations initiales et le principe de conservation :

$$[E_m] = [E_m]_0 - \sum_t \sum_{p \in t} [E_m S_{t,p}] \tag{3.11}$$

$$[S_{t,p}] = [S_t] - \sum_{p' | p \in \mathbf{F}(p')} [E_m S_{t,p'}] \tag{3.12}$$

À noter que l'équation 3.12 ne donne pas la concentration d'un site d'interaction, mais bien d'une position individuelle. Pour calculer celle du site, il faut se référer à l'équation

2.10.

On s'intéresse à deux problèmes : celui d'intégrer numériquement ces équations pour faire évoluer le système dans le temps et celui du calcul de l'état d'équilibre où toutes les équations sont nulles et les contraintes de conservations sont respectées.

3.3 Calcul de l'état d'équilibre

Dans le chapitre 2, nous présentons superficiellement le calcul à effectuer afin d'obtenir la matrice jacobienne nécessaire à la résolution efficace du point stationnaire du système. Nous allons développer ici les expressions analytiques des dérivées et démontrer certaines propriétés de cette matrice.

La méthode de Newton-Raphson est utilisée pour trouver l'état d'équilibre du système, ce qui nécessite d'évaluer deux choses à chaque itération

le vecteur de l'état dynamique du système :

$$\frac{\partial \mathcal{S}}{\partial t} \tag{3.13}$$

la matrice jacobienne :

$$J = \frac{\partial}{\partial \mathcal{S}} \frac{\partial \mathcal{S}}{\partial t} \tag{3.14}$$

La matrice jacobienne contient les dérivées partielles de chaque équation qui caractérisent l'évolution dynamique du système par rapport à chaque variable qui caractérisent l'état $\mathcal{S}$. Il s'agit d'une matrice structurellement symétrique en termes de position des valeurs non-nulles, mais les valeurs correspondantes ne sont pas identiques puisqu'en général, $\frac{\partial}{\partial \mathcal{S}_j} \frac{\partial \mathcal{S}_i}{\partial t} \neq \frac{\partial}{\partial \mathcal{S}_i} \frac{\partial \mathcal{S}_j}{\partial t}$ lorsque $i \neq j$.

L'équilibre est obtenu en trouvant un $\mathcal{S}$ tel que :

$$\frac{\partial \mathcal{S}}{\partial t} = \vec{0} \tag{3.15}$$

On montre également dans le chapitre 2 qu'il n'est pas nécessaire d'incorporer toutes les variables d'états dans la résolution de l'équilibre. En effet, en supposant d'emblée l'existence d'un équilibre, la totalité du système peut être obtenue à partir des concentrations d'équilibre

des complexes $[ES]$ à l'aide des équations de conservation. Par conséquent, le problème équivaut à trouver des valeurs de concentrations de complexes telles que :

$$\frac{\partial [ES]}{\partial t} = \vec{0} \tag{3.16}$$

La méthode de Newton établit la mise à jour suivante pour chaque itération :

$$[ES]^{(t+1)} = [ES]^{(t)} - J^{-1} \frac{\partial [ES]}{\partial t} \tag{3.17}$$

$$\implies -J([ES]^{(t+1)} - [ES]^{(t)}) = \frac{\partial [ES]}{\partial t} \tag{3.18}$$

Le système est réorganisé de sorte à obtenir une forme $Ax = b$ (voir équation 3.18) pouvant se résoudre efficacement avec la méthode de factorisation LU sans nécessiter une inversion explicite de J.

Maintenant, nous allons dériver chaque composante de $\frac{\partial [ES]}{\partial t}$ par rapport à la concentration de chaque complexe du système afin de pouvoir facilement exprimer la matrice jacobienne.

$$\begin{aligned} \frac{\partial [E_m]}{\partial [E_{m'} S_{t',p'}]} &= \frac{\partial}{\partial [E_{m'} S_{t',p'}]} [E_m]_0 - \sum_t \sum_p [E_m S_{t,p}] \\ &= \begin{cases} -1 & \text{si } m = m' \\ 0 & \text{sinon} \end{cases} \end{aligned}$$

Ici, si $m = m'$, il est garanti d'exister au moins et une seule une paire (t, p) pour laquelle $(t, p) = (t', p')$.

$$\frac{\partial[S_{t,\mathbf{F}(p)}]}{\partial[E_{m'}S_{t',p'}]} = \frac{\partial}{\partial[E_{m'}S_{t',p'}]}\Pr[\mathbf{F}(p) \text{ soient libres}][S_t] \tag{3.19}$$

$$= \frac{\partial}{\partial[E_{m'}S_{t',p'}]}\Pr[p' \text{ soit libre}]\underbrace{\Pr[\mathbf{F}(p) - p' \text{ soient libres} \mid p' \text{ est libre}][S_t]}_{\text{ne font pas référence à}[E_{m'}S_{t',p'}]}$$

$$= \Pr[\mathbf{F}(p) - p' \text{ soient libres} \mid p' \text{ est libre}][S_t]\frac{\partial}{\partial[E_{m'}S_{t',p'}]}\Pr[p' \text{ soit libre}]$$

$$= \Pr[\mathbf{F}(p) - p' \text{ soient libres} \mid p' \text{ est libre}][S_t]\frac{\partial}{\partial[E_{m'}S_{t',p'}]}\frac{[S_{t,p'}]}{[S_t]}$$

$$= \begin{cases} -\Pr[\mathbf{F}(p) - p' \text{ soient libres} \mid p' \text{ est libre}] & \text{si } \mathbf{F}(p) \cap \mathbf{F}(p') \neq \emptyset \\ 0 & \text{sinon} \end{cases} \tag{3.20}$$

À l'étape 3.19, on utilise astucieusement le fait qu'on peut choisir arbitrairement la factorisation de la distribution jointe pour séparer les termes contenant p' de ceux qui ne le contiennent pas.

Pour l'étape 3.20, il faut se référer à la définition du modèle :

$$[S_{t,p'}] = [S_t] - \sum_{p|p' \in \mathbf{F}(p)} \sum_m [E_m S_{t,p}] \tag{3.21}$$

Si $p' \in \mathbf{F}(p)$, le terme $[E'_m S_{t',p'}]$ apparaît une seule fois dans l'expression de conservation et sa dérivée sera -1.

On se permet de faire une approximation ici pour éviter de calculer les dérivées des complexes à l'intérieur de l'empreinte :

$$\frac{\partial[S_{t,\mathbf{F}(p)}]}{\partial[E_{m'}S_{t',p'}]} \approx \begin{cases} -1 & \text{si } p = p' \\ 0 & \text{sinon} \end{cases} \tag{3.22}$$

Pour les concentrations de complexes, on aura :

$$\frac{\partial[E_m S_{t,p}]}{\partial[E_{m'}S_{t',p'}]} = \begin{cases} -1 & \text{si } m = m',\ t = t' \text{ et } p = p' \\ 0 & \text{sinon} \end{cases}$$

Le terme k_{autre} introduit des termes de second ordre de par sa définition.

$$\frac{\partial k_{autre}}{\partial [E_{m'} S_{t',p'}]} = \frac{\partial}{\partial [E_{m'} S_{t',p'}]} \sum_{m''} \sum_{p'' | \mathbf{F}(p'') \cap \mathbf{F}(p') = \emptyset} \frac{k_{cat} [E_{m''} S_{t,p''}]}{[S_t]}$$

En recombinant tous les termes, on obtient une dérivée de chaque complexe par rapport aux autres.

$$\begin{aligned}
\frac{\partial}{\partial [E_{m'} S_{t',p'}]} \frac{\partial [E_m S_{t,p}]}{\partial t} &= \frac{\partial}{\partial [E_{m'} S_{t',p'}]} k_f [E_m][S_{t,\mathbf{F}(p)}] - (k_r + k_{cat} + k_{autre})[E_m S_{t,p}] \\
&= k_f ([E_m] \frac{\partial [S_{t,\mathbf{F}(p)}]}{\partial [E_{m'} S_{t',p'}]} + [S_{t,\mathbf{F}(p)}] \frac{\partial [E_m]}{\partial [E_{m'} S_{t',p'}]}) \\
&\quad - (k_r + k_{cat} + k_{autre}) \frac{\partial [E_m S_{t,p}]}{\partial [E_{m'} S_{t',p'}]} \\
&\quad - \frac{\partial k_{autre}}{\partial [E_{m'} S_{t',p'}]} [E_m S_{t,p}]
\end{aligned}$$

Il est clair que la matrice J est très éparse puisque ses entrées non nulles correspondent aux paires d'équation-complexe qui partagent le même microARN ou le même substrat. On distingue deux cas spécifiques pour le partage du substrat : via l'empreinte lorsque $\mathbf{F}(p) \cup \mathbf{F}(p') \neq \emptyset$ ou bien via k_{autre} lorsqu'ils s'hybrident indépendamment.

3.4 Propriétés utiles de l'état du système

Nous montrons dans cette section quelques mesures utiles à calculer et permettant de faire une interprétation éclairée de l'état du système qu'il soit à l'équilibre ou non.

Nombre espéré de microARN occupant une unité de substrat

Soit X, le nombre de microARN occupant une cible donnée S_t et l'ensemble des complexes formés sur cette cible $[E_m S_{t,p}]$, le nombre espéré d'occupants est obtenu en sommant les probabilités de la distribution Poisson-Binomiale :

$$\mathbb{E}[X] = \sum_p \sum_m \frac{[E_m S_{t,p}]}{[S_t]} \tag{3.23}$$

Fraction liée de l'ARN cible

La fraction liée indique la proportion du substrat d'ARN qui est régulé par au moins un microARN. En assumant l'indépendance des sites d'hybridation, elle correspond au complément du cas $\Pr[k = 0]$ de la distribution Poisson-Binomiale.

Soient un ARN cible S_t possédant des sites identifiés par p et un ensemble d'interactions $E_m S_{t,p}$ entre ces sites et l'ensemble des microARN présents dans le système. La fraction des cibles attachée à au moins un microARN est donné par :

$$\mathrm{f}_{\text{liée}}(S_t) = 1 - \prod_{p \in t} \left[1 - \frac{\sum_m [E_m S_{t,p}]}{[S_t]} \right] \tag{3.24}$$

Efficacité d'une interaction enzyme-substrat

L'efficacité est mesurée en prenant le rapport entre la concentration d'un complexe enzyme-substrat à l'équilibre en résolvant les équations de miRBooking 2.0 et la concentration dans un système où seulement l'enzyme et le substrat se retrouvent.

Soient les concentrations initiales d'enzyme $[E]_0$ et de substrat $[S]_0$, la solution à l'équilibre du complexe formé dans un tel système où seuls ces deux espèces peuvent interagir est obtenue en résolvant une équation quadratique lorsque $\frac{\partial [ES]}{\partial t} = 0$:

$$[ES]_{\max} = \frac{Z - \sqrt{Z^2 - 4[E]_0[S]_0}}{2} \qquad \text{avec } Z = [E]_0 + [S]_0 + K_m \tag{3.25}$$

Cette concentration correspond aussi à sa valeur maximale à l'équilibre puisqu'elle est issue d'un système idéal où il n'y a pas de compétitions avec d'autres enzymes et substrats.

Soit la concentration d'un complexe enzyme-substrat $[ES]$ et les concentrations initiales d'enzymes $[E]_0$ et de substrat $[S]_0$, l'efficacité $\eta_{[ES]}$ est donnée par :

$$\begin{aligned} \eta_{[ES]} &= \frac{[ES]}{[ES]_{\max}} \\ &= \frac{2[ES]}{Z - \sqrt{Z^2 - 4[E]_0[S]_0}} \qquad \text{avec } Z = [E]_0 + [S]_0 + K_m \end{aligned} \tag{3.26}$$

Lorsqu'elle est proche de 1, l'interaction enzyme-substrat est très spécifique et peu influencée par les facteurs extérieurs comme la compétition ou la dilution. Elle n'est pas indicatrice

du degré d'activité régulatrice puisqu'un complexe de faible affinité pourrait en principe être très efficace.

Fraction traduite de l'ARN messager

Pour les ARN messagers codants, on s'intéresse à savoir la proportion des molécules qui sont traduites en protéine. En se basant sur les résultats de Mukherji et al. [55], nous proposons une relation log-linéaire entre le nombre de sites occupés k et la fraction traduite.

$$\mathrm{f}_{\mathrm{traduite}}(S_t, k) = e^{-\lambda_1 k + \lambda_0} \tag{3.27}$$

Nous savons que le point d'interception λ_0 est nulle puisqu'en excluant les facteurs extérieurs qui peuvent influencer la traduction d'un ARN, sa totalité est traduite lorsque $k = 0$. Cela permet de simplifier la formule par :

$$\mathrm{f}_{\mathrm{traduite}}(S_t, k) = e^{-\lambda} \tag{3.28}$$

La valeur du paramètre λ est estimé à 0.27 sur les données de Mukherji et al. [55].

Par contre, cette formule ne donne que la fraction traduite pour un k donné et pour pouvoir calculer la fraction traduite d'un ensemble de molécules d'ARN messagers, il faut regarder la distribution de tous les cas discrets possibles de k.

Le modèle Poisson-Binomial nous permet d'estimer la probabilité d'observer un certain nombre d'occupants k sur une molécule donnée de l'ARN cible.

Étant donné une molécule d'ARN possédant n sites occupés à diverses proportions $p_1, p_2, ..., p_n$ et que le nombre de sites occupés X suit une distribution $X \sim$ Poisson-Binomial$(n, p_1, ..., p_n)$, nous calculons la fraction traduite de l'ensemble des molécules en prenant la valeur espérée de la fraction traduite sur tous les cas discrets :

$$\mathrm{f}_{\mathrm{traduite}\forall \mathrm{k}}(S_t) = \sum_{k=0}^{n} \Pr[X = k] e^{-\lambda k} \tag{3.29}$$

Dernièrement, en combinant la fraction traduite de l'ARN et sa concentration $[S_t]$, nous obtenons sa concentration traduite :

$$[S_t]_{\text{traduite}} = [S_t] \times \sum_{k=0}^{n} \Pr[X = k] e^{-\lambda k} \tag{3.30}$$

Cette mesure permet entre autres de prédire l'effet qu'aurait l'activité régulatrice des microARN sur la traduction de l'ARN en protéine.

3.5 Extension aux réseaux de protéines

Le rôle principal de la régulation post-transcriptionnelle est d'affiner le processus de traduction de l'ARN messager en protéine. Il est alors intéressant de se demander comment cette régulation affecte le taux de traduction sous l'hypothèse d'un état stationnaire.

Soit un transcrit S_t qui se traduit en protéine Q_t, la variation de la concentration de ce dernier suit l'équation différentielle suivante :

$$\frac{\partial [Q_t]}{\partial t} = k_{trad}[S_t] - k_{deg}[Q_t] \tag{3.31}$$

où k_{trad} est la constante du taux de traduction du transcrit S_t en protéine Q_t et k_{deg} est la constante de dégradation propre à la protéine Q_t.

Sous l'hypothèse d'un état stationnaire, on peut réécrire l'équation sous la forme suivante puisque $\frac{\partial [Q_t]}{\partial t}$ est nul :

$$k_{trad} = k_{deg} \frac{[Q_t]}{[S_t]} \tag{3.32}$$

Il y a quelque chose de très élégant dans cette formulation : tous les termes à droite peuvent être mesurés expérimentalement. En particulier, Mathieson et al. [60] ont mesuré les demi-vies de 4,000 à 6,000 protéines dans 6 modèles de cellule sénescentes. Dans ce modèle biologique, les cellules ne se divisent plus et préservent un état généralement stationnaire de sorte que les taux de traduction et de dégradation ne sont pas influencés par la division cellulaire. Les concentrations peuvent être estimées par spectrométrie de masse et séquençage à haut débit. La deuxième propriété est que k_{trad} sont indépendantes des concentrations d'ARN et de protéines : il dépend essentiellement de l'efficacité de traduction.

Il suffit donc de pouvoir établir un modèle de régression pour k_{trad} en utilisant l'information de la séquence de l'ARN et des prédictions de miRBooking 2.0. Un logarithme permet

de séparer les facteurs multiplicatifs et leurs erreurs correspondantes dans un modèle linéaire avec un a priori gaussien.

$$\log k_{trad} = \log k_{deg} + \log[Q_t] - \log[S_t] + \mathcal{N}(0,1) \tag{3.33}$$

Pour cette tâche, nous proposons un modèle de réseau de neurones récurrent de type LSTM qui lit la séquence d'ARN et les annotations d'activité des microARN prédites et qui, en sortie, émet le logarithme de constante k_{trad} correspondante. Puisque k_{trad} est une constante de taux, elle ne devrait en aucun point dépendre de la quantité d'ARN présente, ce qui nous permet de fournir une mesure de l'activité régulatrice relative telle que la fraction liée et d'isoler toute l'information quantitative à droite de l'équation.

Cette direction de recherche est proposée comme suite à ce projet.

Chapitre 4

miRBooking-scan

miRBooking-scan est une plateforme Web facilitant la consultation et l'interprétation des prédictions du modèle miRBooking 2.0 disponibles à l'adresse https://major.iric.ca/~poirigui/mirbooking-scan/.

Elle offre des prédictions pour un large éventail de données de lignées cellulaires et tissus provenant principalement du projet ENCODE [61].

TAB. 4.1 : Tableau récapitulatif des lignées cellulaires et tissues hébergés sur miRBooking-scan et de leurs identifiants de référence sur ENCODE.

Échantillon	Description	Épigénome (ENCODE)
A549	Adénocarcinome de l'épithélium alvéolaire basal	ENCSR809EFN
AG04450	Fibroblaste pulmonaire	ENCSR604KIW
Lymphocyte B	Cellules du système immunitaire	ENCSR682AXR
BJ	Fibroblaste du prépuce	ENCSR865DMB
Coeur		ENCSR464TTP
Foie		ENCSR228KEB
Gastrocnemius medialis		ENCSR051NBM
		ENCSR120CKY
Glande surrénale		ENCSR209IBA
		ENCSR127KVK
Glande thyroïde		ENCSR172GTQ
		ENCSR066FYC
GM12878	Lymphocytes B	ENCSR447YYN
GM23248	Fibroblastes primaires	ENCSR300AUZ

Échantillon	Description	Épigénome (ENCODE)
GM23338	Cellules souches pluripotentes	ENCSR769ZAH
H1-hESC	Cellules souches embryonnaires	ENCSR820QMS
H9-derived	Cellules du foie	ENCSR217ARS
H9-derived	Cellules neurales progénitrices	ENCSR372BDU
H9-derived	Cellules musculaires lisses	ENCSR116JEF
HeLa-S3	Cancer cervical	ENCSR068MRQ
HepG2	Carcinome des hépatocytes	ENCSR888GEN
K562	Leucémie myéloïde	ENCSR612NLL
Karpas-422	Lymphome non-Hodgkinien des cellules B	ENCSR888PLC
Keratinocyte	Majorité des cellules formant l'épiderme	ENCSR193SZM
MCF-7	Cancer du sein	ENCSR247DVY
OCI-LY7	Échantillon sanguin d'un lymphome non-Hodgkinien des cellules B	ENCSR507JGJ
Utérus		ENCSR700CXT
Vagin		ENCSR069QGS

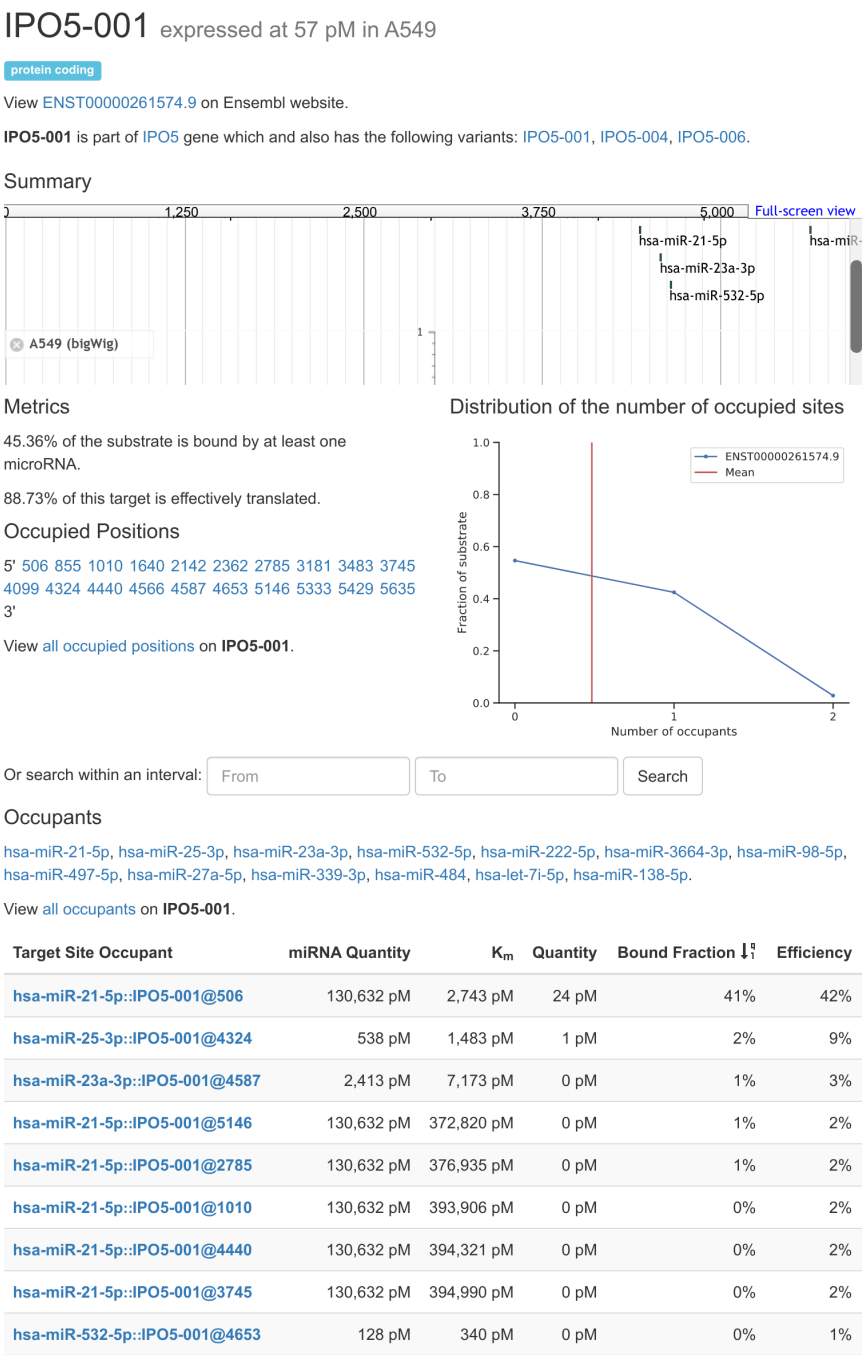
4.1 Implémentation

La plateforme est entièrement écrite en Python à l'aide du cadriciel Flask et du langage Jinja2 pour générer les pages en HTML5. Les prédictions de microtargetome traitées à l'aide de Pandas et SQLite.

Pour assurer un accès efficace aux données, les modèles prédits pour chaque lignée cellulaire sont tout d'abord stockés dans une base de données SQLite. Un index plein texte de recherche est également construit afin de pouvoir trouver facilement les gènes, transcrits et microARN à partir de leurs noms usuels.

Le service Web est adapté pour les accès programmés en offrant les données dans un format tabulaire lorsque l'en-tête `Accept: text/tab-separated-values` est fourni dans la requête.

La solution finale est déployée par CGI en utilisant le module `wsgiref` de la librairie standard Python puisque Flask est compatible avec le protocole WSGI.



Target Site Occupant	miRNA Quantity	K_m	Quantity	Bound Fraction	Efficiency
hsa-miR-21-5p::IPO5-001@506	130,632 pM	2,743 pM	24 pM	41%	42%
hsa-miR-25-3p::IPO5-001@4324	538 pM	1,483 pM	1 pM	2%	9%
hsa-miR-23a-3p::IPO5-001@4587	2,413 pM	7,173 pM	0 pM	1%	3%
hsa-miR-21-5p::IPO5-001@5146	130,632 pM	372,820 pM	0 pM	1%	2%
hsa-miR-21-5p::IPO5-001@2785	130,632 pM	376,935 pM	0 pM	1%	2%
hsa-miR-21-5p::IPO5-001@1010	130,632 pM	393,906 pM	0 pM	0%	2%
hsa-miR-21-5p::IPO5-001@4440	130,632 pM	394,321 pM	0 pM	0%	2%
hsa-miR-21-5p::IPO5-001@3745	130,632 pM	394,990 pM	0 pM	0%	2%
hsa-miR-532-5p::IPO5-001@4653	128 pM	340 pM	0 pM	0%	1%

FIG. 4.1 : L'interface Web de miRBooking-scan permet de visualiser de l'activité régulatrice des microARN sur IPO5-001, un isoforme codant du gène IPO5. On peut voir en haut une piste de la séquence de IPO5-001 annotée par les positions des sites où les microARN se lient, à gauche quelques métriques utiles et à droite la fonction de masse de la distribution Poisson-Binomiale. Une liste exhaustive des interactions prédites se trouve en bas et permet de savoir exactement la concentration, la constante de Michaelis-Menten, la fraction liée du substrat et l'efficacité de chaque microARN.

4.2 Pipeline d'analyse de données de séquençage

Chaque modèle nécessite plusieurs étapes de calculs et d'invocations d'outils bio-informatiques. Tous les traitements effectués sont décris formellement à l'aide du langage WDL [50] et distribués sur une infrastructure de calcul de haute performance à l'aide de Cromwell.

Traitement des données de RNA-Seq

Les données de RNA-Seq sont traitées comme suit :

1. alignement des lectures (ENCODE) ;
2. quantification d'isoformes (ENCODE) ;
3. sélection des cibles de microARN.

Nous considérons que tout ce qui est obtenu lors du séquençage de longs ARN – plus de 200 nucléotides de long – constitue une cible potentielle. Le seul traitement que nous appliquons est de retirer les ARN ribosomaux puisque les échantillons sont quasi systématiquement traités pour en réduire l'abondance.

Traitement des données de microRNA-Seq

Les données de microRNA-Seq sont traitées comme suit :

1. tronquage des adaptateurs (ENCODE) ;
2. alignement des lectures (ENCODE) ;
3. quantification des microARN matures.

Les molécules d'ARN sont traitées avant le séquençage en ajoutant des adaptateurs et des amorces afin de pouvoir les amplifier par PCR. Les microARN matures sont très courts – entre 18 et 22 nucléotides – et les lectures que l'appareil effectue peuvent faire 50 nucléotides de long, ce qui a pour conséquence de lire une partie de l'adaptateur à l'extrémité 3' de la molécule d'intérêt. L'outil cutadapt [45] est utilisé afin de supprimer ces extrémités et améliorer la qualité de l'alignement.

Une fois alignés, les microARN matures sont quantifiés à l'aide de HTSeq [51].

Les quantifications d'isoformes de gènes et de microARN matures sont ensuite combinés dans un modèle d'expression molaire. Puisque les complexes RISC modélisés sont localisés dans le cytoplasme, il faut pouvoir estimer le nombre de molécules présentes et le volume du cytoplasme d'une cellule.

Calcul du microtargetome

Une fois les conditions initiales obtenues, le microtargetome est calculé à l'aide de l'implémentation de miRBooking 2.0 qui peut paralléliser le calcul sur plusieurs cœurs. Les détails du programme sont fournis dans l'appendice A.

Chapitre 5

miRDesign

Le modèle proposé jusqu'à présent permet de caractériser les concentrations des complexes formés dans le système. Du point de vue expérimental, il serait important de pouvoir concevoir des séquences spécifiques à certaines cibles qui prendrait en compte le contexte cellulaire.

On s'intéresse ici au problème de sélectionner les nucléotides d'un microARN guide de sorte à maximiser la répression induite sur un ensemble de gènes à cibler et minimiser celle induite sur un ensemble à éviter.

```
5' UGAGGUAGUAGGUUGUAUAGUU 3'
   |               |-----| queue (7+ nt.)
   |           |--| supplémentaire (4 nt.)
   |       |-| centrale (3-4 nt.)
   ||-----| amorce (7 nt.)
   | tête (1 nt.)
```

Un microARN typique, qu'on nommera synARN pour microARN synthétique dans un contexte génératif, est constitué des éléments suivants :

- un nucléotide libre en première position nommé tête ;
- 7 nucléotides débutant en deuxième position formant l'amorce ;
- la région centrale de 3 à 4 nucléotides ;
- 4 nucléotides supplémentaires ;
- le reste formant la queue.

L'amorce est déterminante pour la spécificité de l'hybridation. La région centrale entoure le site de clivage entre les nucléotides $g_{10}g_{11}$. La région supplémentaire contribue à la stabilité

du complexe et permet à la région centrale de se positionner pour le clivage du substrat. Les nucléotides de la queue renforcent principalement l'affinité et la stabilité du microARN.

De manière naïve, on pourrait simplement énumérer toutes les séquences possibles, mais le nombre de candidats à explorer devient exponentiel dans le nombre de nucléotides à choisir et nous devons recourir à des méthodes de recherche heuristiques.

5.1 Objectif de l'optimisation

Étant donné un ensemble de gène cibles et non-cibles, on cherche à maximiser l'effet moyen relatif entre ces deux groupes. On mesure cet effet sur chaque cible en considérant la fraction du substrat libre de microARN $R(x)$ telle que définie par le complément de l'équation 3.24.

Puisque notre réponse réside dans l'intervalle continu $[0, 1]$ en fonction de la concentration du microARN, nous la transformons à l'aide de la fonction logistique inverse (voir équation 5.1) afin d'avoir un domaine approprié à l'application d'une moyenne arithmétique.

$$\operatorname{logit}(x) = \log \frac{x}{1-x} \tag{5.1}$$

Ensuite, il suffit de faire la différence entre les moyennes des valeurs prises dans chaque espace transformé.

$$\frac{1}{N_{\text{cible}}} \sum_{t \in \text{cibles}} \operatorname{logit} R(t) - \frac{1}{N_{\text{non-cible}}} \sum_{nt \in \text{non-cibles}} \operatorname{logit} R(nt) \tag{5.2}$$

5.2 Heuristique

En s'appuyant sur le modèle de Yan et al. [19], on choisit les nucléotides dans l'ordre suivant :

1. l'amorce de $g_2..g_8$;
2. la boîte B de $g_{12}..g_{14}$;
3. la boîte C de $g_{15}..g_{17}$;
4. la boîte A de $g_9..g_{11}$;
5. le reste des nucléotides de la queue de g_{18} jusqu'à la fin du guide.

Puisque chaque choix effectué dans ces étapes requiert la complémentarité des nucléotides précédents, le rendement sur l'objectif de l'optimisation est décroissant. En projetant linéairement les gains jusqu'à présent sur le reste des choix à faire, on s'assure d'avoir une heuristique admissible puisqu'on sous-estime toujours le coût en surestimant le gain attendu.

5.3 Implémentation

L'implémentation de l'algorithme de recherche est faite à l'aide d'une file de priorité qui ordonne les candidats à traiter en fonction du gain estimé par l'heuristique $g(c) + h(c)$. Un ensemble de fils d'exécutions consomment les candidats et enfilent leurs successeurs (voir algorithme 1).

Algorithm 1 Recherche heuristique parallèle pour trouver le meilleur microARN candidat pour un ensemble de gènes cibles.

Require : c_{init}, un candidat initial
  Soit Q une file de priorités
  Soit c_{best} le meilleur candidat trouvé initialisé à une valeur sentinelle
  Soit T un ensemble de fils d'exécutions
  Insérer le candidat initial c_{init} dans Q
  for all $t \in T$ **do**
    À faire en parallèle
    repeat
      Prendre un candidat c dans Q
      if $g(c) < g(c_{best})$ **then**
        $c_{best} := c$
      end if
      $terminal := true$
      for all $s \in successeurs(c)$ **do**
        if $g(s) + h(s) < g(c_{best})$ **then**
          Ajouter s dans Q avec la priorité $g(s) + h(s)$
          $terminal := false$
        end if
      end for
      if *terminal* **then**
        Émettre c comme candidat terminal
      end if
    until Q soit vide.
  end for
  Attendre que chaque fils d'exécutions de T terminent
  return c_{best}

L'algorithme est entièrement implanté en Python et le parallélisme se fait présentement via des appels de sous-processus `mirbooking`.

```
mirdesign --target
          --non-target
          --initial-candidate NNNNNNNNNNNNNNNNNNNNN
          --workers 32
          --threads-per-worker 1
          --designs-output-dir designs/
          --mirbooking mirbooking
          [MIRBOOKING_ARGS...]
```

L'usager fournit en ensemble de transcrits cibles via `--target` et non-cible via `--non-target` ainsi que tous les arguments à passer à miRBooking afin de paramétrer le modèle qui sera utilisé pour évaluer les candidats.

Il est également possible de définir l'état initial à explorer à l'aide de `--initial-candidate`, ce qui s'avère très pratique pour développer des sous-arbres élagués par l'heuristique.

Le parallélisme peut être ajusté pour avoir une balance entre le nombre de threads alloués par sous-processus et le nombre total de sous-processus à lancer.

Pour ce projet, le prochain objectif est de réimplanter la file de priorité avec MPI afin de pouvoir distribuer le travail à faire sur plusieurs ordinateurs.

5.4 Application : cibler des marqueurs connus de l'EMT

En collaboration avec Roqaya Imane du laboratoire du Dr. François Major, nous avons appliqué la méthode miRDesign pour concevoir des séquences de microARN synthétiques capables de cibler des éléments importants de la transition épithélio-méshenchymale (EMT). Cette transition est très importante dans le développement du cancer puisqu'elle permet aux cellules cancéreuses de métastaser, c'est-à-dire de se répandre pour former d'autres tumeurs.

Tab. 5.1 : Liste des gènes à cibler pour dans le projet d'EMT.

Marqueur	Description
CXCR4	Récepteur membranaire
SCARA5	Récepteur médiateur de la livraison du fer
SNAI1	Facteur de transcription

Marqueur	Description
TLE1	Facteur de transcription
TBK1	Kinase
TWIST	Facteur de transcription
ZEB1	Facteur de transcription
CDH1 (non-cible)	Protéine importante pour l'adhérence des cellules

L'objectif préliminaire consiste à cibler des marqueurs connus (voir tableau 5.1) dans le modèle de cellule A549 afin de valider la méthode et le protocole expérimental. Les meilleurs modèles prédits ont déjà été acquis sous la forme d'oligonucléotides et sont en cours de test (voir figure 5.2).

L'objectif principal du projet qui est encore en cours de réalisation est d'appliquer cette méthodologie sur des cibles inconnues qui seront identifiées par une analyse différentielle entre les deux conditions épithéliale et mésenchymateuse.

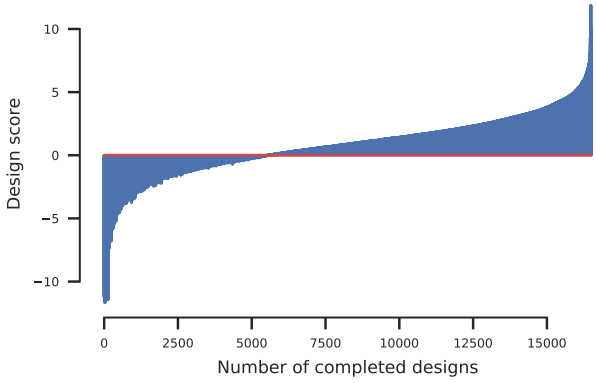


FIG. 5.1 : Progression des scores des candidats obtenus en fonction du nombre de modèles complétés.

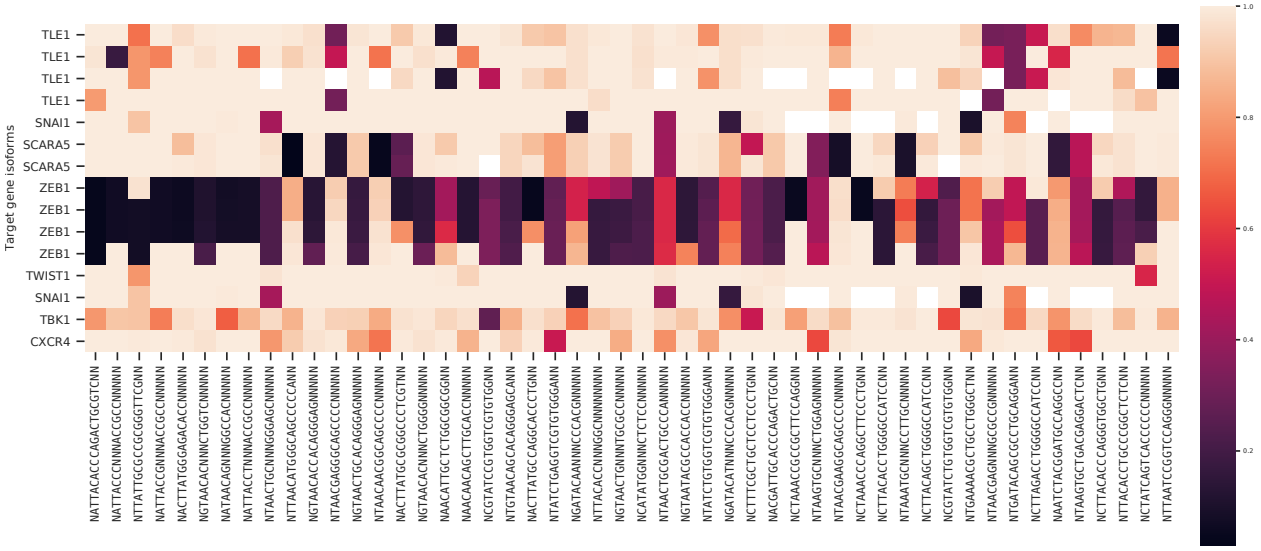


FIG. 5.2 : Résumé des meilleurs candidats obtenus par la méthode miRDesign. L'intensité de la couleur indique la fraction liée entre le candidat et les gènes cibles relative à celle des gènes non-cibles. Une case noire indique que le candidat lie quasi-absolument toute sa cible tandis qu'une case blanche indique que le candidat est non-spécifique.

Chapitre 6

Conclusions et perspectives

Dans le chapitre 1, nous introduisons les notions de biologie nécessaire pour comprendre la nature du problème en passant en revue les notions d'ARN, de microARN, des propriétés des réactions biochimiques et du calcul de leur état d'équilibre. Nous discutons également des éléments importants de la méthode originale de miRBooking développée par Weill et al. [22] afin de la mettre en contraste le travail présenté dans ce mémoire.

Dans le chapitre 2, nous proposons une méthode novatrice basée sur un modèle enzymatique et l'équation de Michaelis-Menten pour modéliser la dynamique des interactions entre les complexes RISC et leurs ARN cibles. En bref, ce modèle résume les complexes et leurs cibles par un système d'enzyme-substrat. Les propriétés biochimiques sont déterminées par un modèle d'interaction interchangeable basé sur plusieurs jeux de données expérimentales. À l'aide des concentrations initiales, nous calculons les concentrations d'équilibre des complexes et modélisons l'évolution temporelle d'un système suite à un changement des conditions expérimentales (p. ex. l'introduction d'un microARN dans le cadre d'une thérapie génique).

Nous démontrons la nature éparse de la jacobienne du système en exhibant les conditions suffisantes pour qu'une entrée soit non-nulle et en montrant qu'en pratique ces cas ne sont pas satisfaits pour la très grande majorité des paires de complexes RISC::cible. Cette propriété nous permet de traiter efficacement des instances de grande taille.

Dans la section 3.5, nous proposons une extension au modèle biochimique pour associer l'activité régulatrice des microARN et l'effet sur la protéine codée par que le gène ciblé. On pose tout d'abord un modèle simple de traduction où le taux de production de la protéine est proportionnel à la quantité d'ARN disponible. On déduit que ce taux est une combinaison log-linéaire du taux de dégradation et des concentrations d'équilibres d'ARN messager et de protéine. Par un modèle d'apprentissage profond, nous suggérons qu'il soit possible d'inférer le taux de dégradation propre à une combinaison de microARN occupants une cible donnée.

Puisque la protéine et sa conformation spatiale confère la fonction ultime d'un gène, la

possibilité de modéliser l'impact d'un microARN sur sa concentration ouvre la porte au design de thérapies géniques efficaces.

Finalement, nous faisons un lien avec un exemple concret où ce type de thérapie génique permet d'avoir un impact positif sur la santé humaine en ciblant des éléments de la transition épithélio-mésenchymateuse, un mécanisme permettant au cancer de proliférer. Pour ce faire, nous proposons une méthode heuristique permettant d'explorer l'espace de la séquence du microARN guide avec pour objectif de maximiser la réponse sur un ensemble de gène cible.

Il est important de réitérer que le modèle présenté dans ce travail ne prétend pas offrir une réponse définitive au problème que présente la régulation génique. L'activité des microARN constitue une partie important certes, mais encore très peu comprise et de nombreuse hypothèse de travail dû être mise en place afin de se concentrer sur un aspect particulier : la notion d'équilibre global du système. Quoique nous ayons avec succès obtenu une réponse affirmative en ce qui concerne la praticité possibilité de résoudre de très grands systèmes, nous sommes encore loin d'un modèle pouvant offrir des prédictions suffisamment précises pour être utiles. La prochaine étape consiste donc à coupler notre approche avec un modèle d'interaction qui reflète mieux la réalité biologique. Plusieurs travaux publiés récemment dont les mesures sur plus de 40,000 interactions par Becker et al. [11] portent à penser que les progrès futurs seront rapides et que nous pourront bientôt avoir toutes les pièces pour compléter ce casse-tête.

Annexe A

Détails d'implémentation de miRBooking 2.0

Puisqu'une grande partie du travail effectué porte sur l'implémentation de la méthode décrite dans le chapitre 2, ce chapitre est dédié aux diverses contributions techniques incorporée dans le programme.

A.1 Organisation du programme

L'organisation du programme repose sur le paradigme orienté-objet et subdivise les différentes composantes dans les classes décrites dans le tableau A.1 de la page 57. Tout ceci est rendu possible grâce à GObject, une librairie permettant un style de programmation orienté-objet en C.

Tab. A.1 : Résumé des classes et leurs responsabilités dans l'implémentation de miRBooking 2.0.

Classe	Responsabilités
`Broker`	Manipulation de l'état du système (c.-à-d. définition des conditions initiales, intégration numérique, résolution de l'équilibre, obtention de la solution)
`ScoreTable`	Fournit les propriétés biochimiques des interactions
`DefaultScoreTable`	Implémentation de référence de la table de score
`Sequence`	Manipulation des séquences biologiques
`Target`	Manipulation d'une séquence d'ARN cible
`Mirna`	Manipulation d'une séquence d'un microARN

La classe abstraite `ScoreTable` est particulièrement intéressante, car elle permet de détacher le modèle biochimique du reste du programme. En pratique, cela permet de modifier très facilement le système en fournissant une définition personnalisée de ces valeurs. Une implémentation fournit deux méthodes : `compute_positions` obtient la liste des positions où les interactions sont possibles et `compute_score` donne les propriétés biochimiques d'une interaction particulière sous la forme d'une structure `Score`. La structure `Score` quant à elle contient les constantes de taux spécifique à la réaction.

L'implémentation par défaut de `compute_positions` traverse toutes les positions de l'ARN cible et vérifie si `compute_score` retourne une constante de dissociation finie $K_d = k_r/k_f$ pour ajouter l'indice correspondant dans la liste des positions.

```
int
score_table_compute_positions (ScoreTable  *self,
                               Target      *target,
                               size_t       position,
                               Mirna       *mirna,
                               size_t     **positions,
                               size_t      *positions_len);

int
score_table_compute_score     (ScoreTable  *self,
                               Target      *target,
                               size_t       position,
                               Mirna       *mirna,
                               Score       *score);
```

En plus de ces classes, un ensemble de structures (voir tableau A.2) permet de représenter et parcourir l'état du système.

Tab. A.2 : Résumé des structures utilisées dans l'implémentation de miRBooking 2.0.

Structure	Description
`TargetSite`	Représente un site d'hybridation
`Occupant`	Représente un occupant sur un site
`Score`	Représente les propriétés biochimiques d'une interaction

```
struct TargetSite
```

```
{
    Target   *target;
    size_t    position;
    Occupant *occupants;
    size_t    occupants_len;
}

struct Occupant
{
    Mirna *mirna;
    Score  score;
}

struct Score
{
    double kf;
    double kr;
    double kcat;
}
```

En mémoire, les sites d'hybridation `TargetSite` sont contigus pour un ARN cible donné, ce qui permet de parcourir facilement le voisinage d'un site afin de calculer, par exemple, sa concentration libre. Les instances de la structure `Occupant` sont également alignés dans un même bloc de mémoire, permettant de parcourir les interactions très efficacement.

La classe `Broker` est la pièce maîtresse l'interface de programmation puisqu'elle permet de manipuler les concentrations des séquences, progresser vers un état d'équilibre, simuler l'évolution du système dans le temps et inspecter la solution à tout moment. Elle offre une routine hybride pour simuler le système par intégration numérique ou avancer efficacement vers l'état d'équilibre à l'aide de la routine `mirbooking_broker_step` et de l'argument `step_mode` qui permet d'alterner entre ces deux modes.

```
enum MirbookingBrokerStepMode
{
    INTEGRATE,
    SOLVE_STEADY_STATE
}
```

```
int
mirbooking_broker_step (MirbookingBroker         *broker,
                        MirbookingBrokerStepMode  step_mode,
                        double                    step_size);
```

À tout moment, il est possible de déterminer si le système est à l'équilibre. Pour cela, on calcule la norme L_∞ du ratio erreur-tolérance. La tolérance est obtenue en prenant une combinaison d'erreur relative et absolue sur chaque composante du système.

$$\max_{s \in \mathcal{S}} \frac{\frac{\partial s}{\partial t}}{r_{tol} \frac{\partial S}{\partial t} + a_{tol}} \tag{A.1}$$

L'interface de programmation ci-haut est exposé dans une bibliothèque partagée qui est notamment utilisée par le programme principal en ligne de commande.

A.2 Ligne de commande

La méthode proposée est avant tout destinée à être utilisée via la ligne de commande de sorte qu'elle puisse être facilement intégrée à un pipeline bio-informatique.

Voici la liste des arguments supportés :

```
mirbooking --targets gencode.v24.transcripts.fa
           --mirnas mature.fa
           --input quantifications.tsv
           --seed-scores scores-7mer-3mismatch-ending
           --supplementary-model yan-et-al-2018
           --supplementary-scores scores-3mer
           --accessibility-scores gencode.v24.transcripts_accessibility.tsv
           --5prime-footprint 9
           --3prime-footprint 7
           --sparse-solver superlu
           --max-iterations 100
           --cutoff 100
           --output microtargetome.tsv
           --output-format tsv
```

Les arguments `--targets` et `--mirnas` permettent de fournir des fichiers FASTA contenant les séquences de référence des cibles et des microARN.

L'argument `--input` permet de spécifier les quantifications des ARN cibles et des microARN en associant une concentration molaire à l'identifiant unique de la séquence dans un format tabulaire.

Les arguments `--seed-scores` et `--supplementary-scores` permettent de fournir des tables pré-calculée d'énergie libre pour les duplexes ARN-ARN qui sont incorporés dans le modèle thermodynamique.

La sortie du programme peut être dirigé vers un fichier à l'aide de `--output`. Par défaut, le format de sortie est un format tabulaire contenant les colonnes suivantes :

Tab. A.3 : Description du format de sortie tabulaire de miRBooking 2.0

Colonne	Description
`gene_accession`	Identifiant unique du gène qui code pour la cible
`gene_name`	Nom commun du gène
`target_accession`	Identifiant unique de la cible
`target_name`	Nom commun de la cible
`target_quantity`	Concentration de la cible (pM)
`position`	Position sur la cible (commence à 1)
`mirna_accession`	Identifiant unique du microARN
`mirna_name`	Nom commun du microARN
`mirna_quantity`	Concentration du microARN (pM)
`score`	Constante de Michaelis-Menten (pM)
`quantity`	Concentration de l'interaction (pM)

Il est aussi possible de produire des annotations GFF3 [62] et un signal Wiggle afin d'intégrer les prédictions dans un navigateur génomique tel que JBrowse [63].

A.3 Parseur FASTA sans allocations

Les séquences d'ARN sont typiquement stockées dans le format de fichier texte FASTA sous forme d'entrées identifiées de manière unique. Les séquences sont généralement coupées par une nouvelle ligne tous les 80 caractères, quoique cela n'est pas obligatoire. Pour miRBooking, ces fichiers contiennent les séquences d'ARN cibles et de microARN guides et l'opération principale est d'en récupérer de courtes sous-séquences de 3 à 7 nucléotides. De

plus, pour miRDesign, plusieurs modèles doivent être lancés en parallèle afin de déterminer de bonnes séquences guides.

```
>identifiant commentaire
ACTGACTGCCC
TTTACTGACTG
```

En résumé, on cherche à faire de petits accès non-séquentiels dans un fichier qui peut peser près d'un gigaoctet, et ce en consommant le moins de mémoire possible lorsque plusieurs instances du programme seront exécutées. La stratégie qui s'impose est d'exploiter la capacité des systèmes d'exploitation à créer des correspondances entre les fichiers et la mémoire virtuelle de sorte que cette mémoire puisse être implicitement partagée à travers les processus qui l'utilisent. L'appel système utilisé est `mmap` dont la signature est la suivante :

```
void *mmap(void *addr, size_t length, int prot, int flags, int fd,
           off_t offset);
```

Les paramètres `offset` et `length` servent à paramétrer un accès sur une portion du fichier via son descripteur `fd`. L'appel retourne une adresse dans la mémoire virtuelle permettant de faire des accès arbitraires au fichier sous-jacent.

Dans la majorité des accès, c.-à-d. `ACTGAC` en position zéro, la sous-séquence demandée se trouve directement en mémoire et il suffit de calculer correctement son adresse. Pour les autres cas, c.-à-d. `CCCTTT` en huitième position, une nouvelle ligne coupe la sous-séquence en deux, ce qui nécessite deux copies.

La stratégie proposée consiste à préconditionner chaque séquence du fichier FASTA en mémorisant les positions des nouvelles lignes en ordre croissant d'adresses et d'utiliser cette information pour calculer efficacement les adresses effectives à l'aide de l'algorithme 2. L'implémentation utilise un tampon statique par fil d'exécution via le *thread-local storage* afin de gérer les cas d'accès concurrents.

A.4 Initialisation parallèle d'une matrice éparse

Puisque la matrice jacobienne est encodée dans le format Yale, il est nécessaire de produire les rangées de la matrice en ordre croissant. Pour l'initialisation, cela pose un problème de dépendance de données : puisqu'il devient nécessaire de connaître `rowptr[i - 1]` afin de définir `rowptr[i]`.

Pour contourner ce problème, on produit séparément les rangées de la matrice afin de les recombiner par copies séquentielles. Cette stratégie est très efficace dans notre cas, puisque

Algorithm 2 Accès d'une sous-séquence sans effectuer d'allocations.

Require : seq, une séquence de n caractères
Require : linefeeds, un tableau d'indices vers les nouvelles lignes de la séquence seq
Require : offset, length un accès d'un sous-intervalle de seq
  Soit seq_buffer, un tampon mémoire de longueur length
  Soit seq_buffer_offset un indice de progression pour seq_buffer initialisé à 1
  for all linefeed dans linefeeds **do**
    if linefeed précède offset **then**
      offset := offset + 1
    else if linefeed précède offset + length **then**
      Copier seq[offset :linefeed] dans seq_buffer[seq_buffer_offset :]
      seq_buffer_offset := seq_buffer_offset + (linefeed - offset)
    else if seq_buffer_offset == 1 **then**
      return seq[offset :offset+length]
    else
      Copier seq[offset :offset+(length - seq_buffer_offset)] dans seq_buffer[seq_buffer_offset :]
      return seq_buffer
    end if
  end for

Algorithm 3 Initialisation parallèle d'une matrice éparse.

Require : n, la dimension de la matrice et nnz son nombre de valeurs non-nulles
Require : initialize_row, une routine d'intialisation pour la rangée i
  Soit rowptr, un tableau de n valeurs
  Soit colind et data, deux tableaux de taille nnz chacun
  rowptr[1] = 1
  for i := 1 jusqu'à n **do**
    Lancer initialize_row(i) en parallèle
  end for
  for i := 1 jusqu'à n **do**
    row_colind, row_data := attendre initialize_row(i)
    rowptr[$i+1$] = rowptr[i] + taille de row_colind
    colind[rowptr[i] :rowptr[$i+1$]] := row_colind
    data[rowptr[i] :rowptr[$i+1$]] := row_data
  end for
  return rowptr, colind, data

copier les rangées contigües est beaucoup moins coûteux que les produire. On exploite donc mieux nos ressources de calcul en parallélisant la production.

A.5 Intégration numérique explicite

Le module `contrib/odeint` implémente une variété de méthodes d'intégrations numériques explicites à l'aide de tableaux Butcher [64]. Il serait en principe aussi possible d'utiliser des méthodes implicites puisqu'on sait déjà comment calculer la matrice jacobienne, mais le coût par itération deviendrait exorbitant.

Les méthodes explicites d'intégration sont basées sur des expansions particulières de la série de Taylor de la fonction à intégrer. Il existe deux types d'erreurs commises par un intégrateur numérique : l'erreur de troncation locale et l'erreur globale. L'erreur locale correspond à l'ordre du terme tronqué lors de l'expansion de la fonction et l'erreur globale correspond à l'erreur accumulée depuis le début de l'intégration.

Les méthodes explicites ne permettent pas de contrôler directement l'erreur globale, mais certaines proposent une stratégie prédicteur-correcteur et fournissent un terme pour estimer l'erreur de troncation locale. Avec cette information, il est possible d'ajuster le pas en fonction d'une tolérance τ donnée et l'ordre de convergence p de la méthode [65] (voir équation A.2).

$$h_{\text{optimal}} = h_{\text{courant}} \left(\frac{\tau}{\delta} \right)^{\frac{1}{p}} \tag{A.2}$$

Puisqu'on ne se limite pas aux systèmes d'une seule équation dans notre application de cette méthode, nous utilisons une norme L_∞ afin de déterminer l'erreur de troncation locale.

TAB. A.4 : Tableau des méthodes d'intégration numérique explicite supportées par le module `odeint` de miRBooking 2.0. Une méthode est dite intégrée si elle fournit un terme permettant d'estimer l'erreur de troncation locale.

Méthode	Ordre	Nombre d'évaluations	Intégrée
Euler	$O(h)$	1	Non
Heuns	$O(h^2)$	2	Non
Heuns-Euler	$O(h^2)$	3	Oui
Bogacki-Shampine[66]	$O(h^3)$	4	Oui
Runge-Kutta[67]	$O(h^4)$	5	Oui
Runge-Kutta-Fehlberg[68]	$O(h^5)$	5	Oui
Cash-Karp[69]	$O(h^5)$	7	Oui

Méthode	Ordre	Nombre d'évaluations	Intégrée
Dormand-Prince[44]	$O(h^5)$	6	Oui

La méthode de Dormand-Prince comporte un avantage considérable puisqu'elle s'affranchit d'une évaluation de la fonction en réutilisant sa dernière étape comme pas final.

L'intégrateur est initialisé en choisissant une méthode parmi celles décrites dans le tableau A.4, une région de mémoire où t et y seront stockés et mis à jour ainsi qu'une paire de tolérances relative et absolue.

Une fonction intégrable $f(t, y)$ selon t produit directement ses dérivées partielles dans le vecteur F à l'aide d'un pointeur de fonction.

```
OdeIntIntegtator *
odeint_integrator_new (OdeIntIntegratorMethod  method,
                       double                 *t0,
                       double                 *y0,
                       size_t                  n,
                       double                  rtol,
                       double                  atol);

void
odeint_integrator_integrate (OdeIntIntegrator *self,
                             OdeIntFunc        func,
                             void             *user_data,
                             double            tw);
```

A.6 Solveur linéaire épars

miRBooking 2.0 supporte plusieurs solveurs linéaires épars à travers l'interface décrite dans le module `contrib/sparse`.

L'interface de programmation est assez simple puisqu'il nécessite seulement de créer une instance de `SparseSolver` et d'appeler la méthode `sparse_solver_solve` pour résoudre un système de la forme $Ax = b$.

```
SparseSolver *
sparse_solver_new    (SparseSolverMethod solver_method);
```

```
sparse_solver_solve (SparseSolver *solver,
                     SparseMatrix *A,
                     void         *x,
                     void         *b);
```

Les solveurs suivants sont supportés :

- SuperLU et sa variante parallèle SuperLU MT [70] ;
- UMFPACK [71] ;
- Intel MKL PARDISO (via MKL DSS) ;
- Intel MKL Cluster (distribué via MPI) ;
- Intel MKL LAPACK ;
- NVIDIA cuSOLVER ;
- PARDISO.

Un solveur basé sur LAPACK [72] permet de traiter efficacement les cas où la jacobienne est dense. Ce type de cas survient lorsqu'on cherche à caractériser l'ensemble des cibles d'un microARN puisque toutes les réactions partageront le même enzyme.

Pour résoudre les problèmes de très grande taille où le nombre de valeurs non-nulles peut faire déborder un entier sur 32 bits, un mode 64 bits peut être activé à la configuration pour les solveurs qui le supporte.

Par défaut, le meilleur solveur est choisi parmi ceux activés à la configuration en suivant cet ordre pré-établi : MKL DSS, PARDISO, UMFPACK, SuperLU MT, SuperLU et finalement LAPACK.

A.7 Bindings avec GObject Introspection

miRBooking 2.0 est utilisable via Python et JavaScript grâce aux *bindings* générés automatiquement à l'aide de GObject Introspection [73].

```
from gi.repository import Mirbooking

mirna = Mirbooking.Mirna(accession='MIMAT000001', 'ACTG....ACTG')
target = Mirbooking.Target(accession='ENST0000001', sequence='ACTG.....ACTG')

score_table = Mirbooking.DefaultScoreTable(...)
```

```
broker = Mirbooking.Broker(score_table=sc)

broker.set_sequence_quantity(mirna, 1e3)
broker.set_sequence_quantity(target, 1e4)

while True:
    ret, eto = broker.evaluate()
    if eto < 1:
        break
    broker.step(Mirbooking.BrokerStepMode.SOLVE_STEADY_STATE, 1.0)
```

Une fois notre état d'équilibre calculé, les prédictions peuvent être explorées librement. Les prédictions sont parcourues à l'aide de `Broker.get_target_sites`.

```
for target_site in broker.get_target_sites():
    for occupant in target_site.occupants:
        ES = broker.get_occupant_quantity(occupant)
```

Pour éviter de traverser inutilement les positions libres, on utilise directement la routine `Broker.get_occupants`.

```
for occupant in broker.get_occupants():
    ES = broker.get_occupant_quantity(occupant)
```

On calcule la distribution du nombre d'occupants donnée par le modèle Poisson-Binomial via `Broker.get_target_occupants_pmf`.

```
broker.get_target_occupants_pmf(target)
```

A.8 Implémentations de la distribution Poisson-Binomiale

Les deux routines qui suivent initialisent `pb->pmf` qui est un vecteur de `pb->n + 1` valeurs contenant la probabilité d'observer 0 à `pb->n` succès à partir de `pb->p` le vecteur de probabilités pour chaque événement et `pb->n` le nombre d'événements.

```
typedef struct _PoissonBinomial {
  double *p;
```

```
  size_t n;
  double *pmf;
} PoissonBinomial;
```

Par programmation dynamique

L'optimisation pour $T(j)$ n'est pas présentée dans l'extrait de code qui suit.

```
void pb_init (PoissonBinomial *pf)
{
  pb->pmf[0] = 1;
  int i;
  for (i = 0; i < pb->n; i++)
  {
    pb->pmf[0] *= (1 - pb->p[i]);
  }

  int k;
  for (k = 1; k < pb->n + 1; k++)
  {
    double pk = 0;
    int i;
    for (i = 0; i < k; i++)
    {
      // cette boucle peut être précalculée pour tous i en dehors de la boucle
      // sur les k
      double ti = 0;
      int j;
      for (j = 0; j < pb->n; j++)
      {
        ti += pow (pb->p[j] / (1 - pb->p[j]), i + 1);
      }
      pk += (i % 2 == 0 ? 1 : -1) * pb->pmf[k - i - 1] * ti;
    }
    pb->pmf[k] = pk / k;
  }
}
```

Par la transformée de Fourier

La librairie FFTW3 [74] est nécessaire pour les calculs qui suivent.

```
void pb_init (PoissonBinomial *pf)
{
  fftw_complex *in;
  fftw_plan plan;
  size_t N = pb->n + 1;

  in = fftw_malloc (sizeof (fftw_complex) * N);

  plan = fftw_plan_dft_c2r_1d (N,
                               in,
                               pb->pmf,
                               FFTW_ESTIMATE);

  // initialisation
  int n;
  for (n = 0; n < N; n++)
  {
    in[n] = 1.0;

    int k;
    for (k = 0; k < pb->n; k++)
    {
      in[n] *= pb->p[k] * cexp (-I * 2.0 * M_PI * n / N) + (1 - pb->p[k]);
    }
  }

  fftw_execute (plan);

  for (n = 0; n < N; n++)
  {
    pb->pmf[n] /= N;
  }
```

```
  fftw_destroy_plan (plan);
  fftw_free (in);
}
```

Index analytique

Bibliographie

[1] Rosalind C. Lee, Rhonda L. Feinbaum et Victor Ambros. « The C. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14 ». In : *Cell* 75.5 (3 déc. 1993), p. 843-854. issn : 0092-8674, 1097-4172. doi : 10.1016/0092-8674(93)90529-Y. url : https://www.cell.com/cell/abstract/0092-8674(93)90529-Y (visité le 20/07/2019).

[2] Brenda J. Reinhart et al. « The 21-nucleotide let-7 RNA regulates developmental timing in Caenorhabditis elegans ». In : *Nature* 403.6772 (fév. 2000), p. 901. issn : 1476-4687. doi : 10.1038/35002607. url : https://www.nature.com/articles/35002607 (visité le 29/07/2019).

[3] Ana Kozomara, Maria Birgaoanu et Sam Griffiths-Jones. « miRBase : from microRNA sequences to function ». In : *Nucleic Acids Research* 47 (D1 8 jan. 2019), p. D155-D162. issn : 0305-1048. doi : 10.1093/nar/gky1141. url : https://academic.oup.com/nar/article/47/D1/D155/5179337 (visité le 29/07/2019).

[4] Sung Wook Chi et al. « Argonaute HITS-CLIP decodes microRNA–mRNA interaction maps ». In : *Nature* 460.7254 (juil. 2009), p. 479. issn : 1476-4687. doi : 10.1038/nature08170. url : https://www.nature.com/articles/nature08170 (visité le 07/06/2018).

[5] Vikram Agarwal et al. « Predicting effective microRNA target sites in mammalian mRNAs ». In : *eLife* 4 (2015). issn : 2050-084X. doi : 10.7554/eLife.05005. url : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4532895/ (visité le 18/10/2017).

[6] James P. Broughton et al. « Pairing beyond the Seed Supports MicroRNA Targeting Specificity ». In : *Molecular Cell* 64.2 (20 oct. 2016), p. 320-333. issn : 1097-2765. doi : 10.1016/j.molcel.2016.09.004. url : http://www.sciencedirect.com/science/article/pii/S1097276516305214 (visité le 18/05/2017).

[7] Nayi Wang et al. « Single-cell microRNA-mRNA co-sequencing reveals non-genetic heterogeneity and mechanisms of microRNA regulation ». In : *Nature Communications*

10.1 (9 jan. 2019), p. 95. ISSN : 2041-1723. DOI : 10.1038/s41467-018-07981-6. URL : https://www.nature.com/articles/s41467-018-07981-6 (visité le 11/04/2019).

[8] Phillip D ZAMORE et al. « RNAi : Double-Stranded RNA Directs the ATP-Dependent Cleavage of mRNA at 21 to 23 Nucleotide Intervals ». In : *Cell* 101.1 (31 mar. 2000), p. 25-33. ISSN : 0092-8674. DOI : 10.1016/S0092-8674(00)80620-0. URL : http://www.sciencedirect.com/science/article/pii/S0092867400806200 (visité le 13/03/2019).

[9] Liang Meng WEE et al. « Argonaute Divides Its RNA Guide into Domains with Distinct Functions and RNA-Binding Properties ». In : *Cell* 151.5 (21 nov. 2012), p. 1055-1067. ISSN : 0092-8674. DOI : 10.1016/j.cell.2012.10.036. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3595543/ (visité le 13/09/2018).

[10] William E. SALOMON et al. « Single-Molecule Imaging Reveals that Argonaute Reshapes the Binding Properties of Its Nucleic Acid Guides ». In : *Cell* 162.1 (2 juil. 2015), p. 84-95. ISSN : 0092-8674. DOI : 10.1016/j.cell.2015.06.029. URL : http://www.sciencedirect.com/science/article/pii/S0092867415007138 (visité le 07/12/2018).

[11] Winston R. BECKER et al. « High-Throughput Analysis Reveals Rules for Target RNA Binding and Cleavage by AGO2 ». In : *Molecular Cell* (16 juil. 2019). ISSN : 1097-2765. DOI : 10.1016/j.molcel.2019.06.012. URL : http://www.sciencedirect.com/science/article/pii/S1097276519304459 (visité le 02/08/2019).

[12] Thomas F. DUCHAINE et Marc R. FABIAN. « Mechanistic Insights into MicroRNA-Mediated Gene Silencing ». In : *Cold Spring Harbor Perspectives in Biology* 11.3 (3 jan. 2019), a032771. ISSN : , 1943-0264. DOI : 10.1101/cshperspect.a032771. URL : http://cshperspectives.cshlp.org/content/11/3/a032771 (visité le 03/03/2019).

[13] Anastasia KHVOROVA, Angela REYNOLDS et Sumedha D. JAYASENA. « Functional siRNAs and miRNAs Exhibit Strand Bias ». In : *Cell* 115.2 (17 oct. 2003), p. 209-216. ISSN : 0092-8674. DOI : 10.1016/S0092-8674(03)00801-8. URL : http://www.sciencedirect.com/science/article/pii/S0092867403008018 (visité le 01/08/2019).

[14] Mallory A. HAVENS et al. « Biogenesis of mammalian microRNAs by a non-canonical processing pathway ». In : *Nucleic Acids Research* 40.10 (mai 2012), p. 4626-4640. ISSN : 0305-1048. DOI : 10.1093/nar/gks026. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3378869/ (visité le 01/08/2019).

[15] Sihem CHELOUFI et al. « A Dicer-independent miRNA biogenesis pathway that requires Ago catalysis ». In : *Nature* 465.7298 (3 juin 2010), p. 584-589. ISSN : 0028-0836. DOI : 10.1038/nature09092. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2995450/ (visité le 01/08/2019).

[16] Gabriel D. Bossé et Martin J. Simard. « A new twist in the microRNA pathway : Not Dicer but Argonaute is required for a microRNA production ». In : *Cell Research* 20.7 (juil. 2010), p. 735-737. issn : 1748-7838. doi : 10.1038/cr.2010.83. url : https://www.nature.com/articles/cr201083 (visité le 04/03/2019).

[17] Robin C. Friedman et al. « Most mammalian mRNAs are conserved targets of microRNAs ». In : *Genome Research* 19.1 (1er jan. 2009), p. 92-105. issn : 1088-9051, 1549-5469. doi : 10.1101/gr.082701.108. url : http://genome.cshlp.org/content/19/1/92 (visité le 03/08/2019).

[18] Benjamin P. Lewis, Christopher B. Burge et David P. Bartel. « Conserved Seed Pairing, Often Flanked by Adenosines, Indicates that Thousands of Human Genes are MicroRNA Targets ». In : *Cell* 120.1 (14 jan. 2005), p. 15-20. issn : 0092-8674. doi : 10.1016/j.cell.2004.12.035. url : http://www.sciencedirect.com/science/article/pii/S0092867404012607 (visité le 03/08/2019).

[19] Yifei Yan et al. « The sequence features that define efficient and specific hAGO2-dependent miRNA silencing guides ». In : *Nucleic Acids Research* (22 juin 2018). doi : 10.1093/nar/gky546. url : https://academic.oup.com/nar/advance-article/doi/10.1093/nar/gky546/5042794 (visité le 22/06/2018).

[20] Misha Klein et al. « Why Argonaute is needed to make microRNA target search fast and reliable ». In : *Seminars in Cell & Developmental Biology*. Non-coding RNAs in development and disease 65 (1er mai 2017), p. 20-28. issn : 1084-9521. doi : 10.1016/j.semcdb.2016.05.017. url : http://www.sciencedirect.com/science/article/pii/S1084952116301434 (visité le 15/06/2018).

[21] Jessica Sheu-Gruttadauria et al. « Beyond the seed : structural basis for supplementary microRNA targeting by human Argonaute2 ». In : *The EMBO Journal* (26 avr. 2019), e101153. issn : 0261-4189, 1460-2075. doi : 10.15252/embj.2018101153. url : http://emboj.embopress.org/content/early/2019/04/26/embj.2018101153 (visité le 13/05/2019).

[22] Nathanaël Weill et al. « MiRBooking simulates the stoichiometric mode of action of microRNAs ». In : *Nucleic Acids Research* 43.14 (18 août 2015), p. 6730-6738. issn : 0305-1048, 1362-4962. doi : 10.1093/nar/gkv619. url : https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkv619 (visité le 04/02/2017).

[23] D. Gale et L. S. Shapley. « College Admissions and the Stability of Marriage ». In : *The American Mathematical Monthly* 69.1 (1962), p. 9-15. issn : 0002-9890. doi : 10.2307/2312726. url : https://www.jstor.org/stable/2312726 (visité le 12/01/2020).

[24] Leonor Menten et MI Michaelis. « Die kinetik der invertinwirkung ». In : *Biochem Z* 49.333 (1913), p. 5.

[25] Nicole T Schirle et al. « Water-mediated recognition of t1-adenosine anchors Argonaute2 to microRNA targets ». In : *eLife* 4 (11 sept. 2015). Sous la dir. de Phillip D Zamore, e07646. issn : 2050-084X. doi : 10.7554/eLife.07646. url : https://doi.org/10.7554/eLife.07646 (visité le 10/12/2018).

[26] Myung Hyun Jo et al. « Human Argonaute 2 Has Diverse Reaction Pathways on Target RNAs ». In : *Molecular Cell* 59.1 (2 juil. 2015), p. 117-124. issn : 1097-2765. doi : 10.1016/j.molcel.2015.04.027. url : http://www.sciencedirect.com/science/article/pii/S1097276515003093 (visité le 15/12/2018).

[27] Benjamin Haley et Phillip D. Zamore. « Kinetic analysis of the RNAi enzyme complex ». In : *Nature Structural & Molecular Biology* 11.7 (juil. 2004), p. 599-606. issn : 1545-9985. doi : 10.1038/nsmb780. url : https://www.nature.com/articles/nsmb780 (visité le 31/01/2019).

[28] Walt F. Lima et al. « Binding and Cleavage Specificities of Human Argonaute2 ». In : *Journal of Biological Chemistry* 284.38 (18 sept. 2009), p. 26017-26028. issn : 0021-9258, 1083-351X. doi : 10.1074/jbc.M109.010835. url : http://www.jbc.org/lookup/doi/10.1074/jbc.M109.010835 (visité le 17/02/2018).

[29] Nadya Morozova et al. « Kinetic signatures of microRNA modes of action ». In : *RNA* 18.9 (sept. 2012), p. 1635-1655. issn : 1355-8382. doi : 10.1261/rna.032284.112. url : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3425779/ (visité le 13/04/2018).

[30] Chikako Ragan, Michael Zuker et Mark A. Ragan. « Quantitative Prediction of miRNA-mRNA Interaction Based on Equilibrium Concentrations ». In : *PLoS Computational Biology* 7.2 (24 fév. 2011). issn : 1553-734X. doi : 10.1371/journal.pcbi.1001090. url : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3044769/ (visité le 12/06/2018).

[31] Austin E. Gillen et al. « Improvements to the HITS-CLIP protocol eliminate widespread mispriming artifacts ». In : *BMC Genomics* 17 (5 mai 2016), p. 338. issn : 1471-2164. doi : 10.1186/s12864-016-2675-5. url : https://doi.org/10.1186/s12864-016-2675-5 (visité le 26/06/2018).

[32] Ronny Lorenz et al. « ViennaRNA Package 2.0 ». In : *Algorithms for Molecular Biology* 6.1 (24 nov. 2011), p. 26. issn : 1748-7188. doi : 10.1186/1748-7188-6-26. url : https://doi.org/10.1186/1748-7188-6-26 (visité le 13/08/2018).

[33] Sung Wook Chi, Gregory J. Hannon et Robert B. Darnell. « An alternative mode of microRNA target recognition ». In : *Nature Structural & Molecular Biology* 19.3 (mar. 2012), p. 321-327. issn : 1545-9985. doi : 10.1038/nsmb.2230. url : https://www.nature.com/articles/nsmb.2230 (visité le 17/10/2018).

[34] Ulrike Mückstein et al. « Thermodynamics of RNA–RNA binding ». In : *Bioinformatics* 22.10 (15 mai 2006), p. 1177-1182. issn : 1367-4803. doi : 10.1093/bioinformatics/btl024. url : https://academic.oup.com/bioinformatics/article/22/10/1177/236620 (visité le 12/12/2018).

[35] Ole Tange. « GNU Parallel : The Command-Line Power Tool ». In : (), p. 6.

[36] Michael Kertesz et al. « The role of site accessibility in microRNA target recognition ». In : *Nature Genetics* 39.10 (oct. 2007), p. 1278-1284. issn : 1546-1718. doi : 10.1038/ng2135. url : https://www.nature.com/articles/ng2135 (visité le 23/06/2018).

[37] Javier Martinez et Thomas Tuschl. « RISC is a 5 phosphomonoester-producing RNA endonuclease ». In : *Genes & Development* 18.9 (5 jan. 2004), p. 975-980. issn : 0890-9369, 1549-5477. doi : 10.1101/gad.1187904. url : http://genesdev.cshlp.org/content/18/9/975 (visité le 17/05/2019).

[38] Pål Sætrom et al. « Distance constraints between microRNA target sites dictate efficacy and cooperativity ». In : *Nucleic Acids Research* 35.7 (1er avr. 2007), p. 2333-2342. issn : 0305-1048. doi : 10.1093/nar/gkm133. url : https://academic.oup.com/nar/article/35/7/2333/1094167 (visité le 29/11/2018).

[39] M. Fernandez et S. Williams. « Closed-Form Expression for the Poisson-Binomial Probability Density Function ». In : *IEEE Transactions on Aerospace and Electronic Systems* 46.2 (avr. 2010), p. 803-817. issn : 0018-9251. doi : 10.1109/TAES.2010.5461658.

[40] David B. Shear. « Stability and Uniqueness of the Equilibrium Point in Chemical Reaction Systems ». In : *The Journal of Chemical Physics* 48.9 (1er mai 1968), p. 4144-4147. issn : 0021-9606. doi : 10.1063/1.1669753. url : https://aip.scitation.org/doi/abs/10.1063/1.1669753 (visité le 04/04/2018).

[41] Arne De Coninck et al. « Needles : Toward Large-Scale Genomic Prediction with Marker-by-Environment Interaction ». In : *Genetics* 203.1 (1er mai 2016), p. 543-555. issn : 0016-6731, 1943-2631. doi : 10.1534/genetics.115.179887. url : http://www.genetics.org/content/203/1/543 (visité le 17/02/2019).

[42] Fabio VERBOSIO et al. « Enhancing the scalability of selected inversion factorization algorithms in genomic prediction ». In : *Journal of Computational Science* 22 (1er sept. 2017), p. 99-108. ISSN : 1877-7503. DOI : 10.1016/j.jocs.2017.08.013. URL : http://www.sciencedirect.com/science/article/pii/S1877750317301473 (visité le 17/02/2019).

[43] D. KOUROUNIS, A. FUCHS et O. SCHENK. « Toward the Next Generation of Multiperiod Optimal Power Flow Solvers ». In : *IEEE Transactions on Power Systems* 33.4 (juil. 2018), p. 4005-4014. ISSN : 0885-8950. DOI : 10.1109/TPWRS.2017.2789187.

[44] J. R. DORMAND et P. J. PRINCE. « A family of embedded Runge-Kutta formulae ». In : *Journal of Computational and Applied Mathematics* 6.1 (1er mar. 1980), p. 19-26. ISSN : 0377-0427. DOI : 10.1016/0771-050X(80)90013-3. URL : http://www.sciencedirect.com/science/article/pii/0771050X80900133 (visité le 01/03/2019).

[45] Marcel MARTIN. « Cutadapt removes adapter sequences from high-throughput sequencing reads ». In : *EMBnet.journal* 17.1 (2 mai 2011), p. 10-12. ISSN : 2226-6089. DOI : 10.14806/ej.17.1.200. URL : https://journal.embnet.org/index.php/embnetjournal/article/view/200 (visité le 27/07/2019).

[46] Tom Sean SMITH, Andreas HEGER et Ian SUDBERY. « UMI-tools : Modelling sequencing errors in Unique Molecular Identifiers to improve quantification accuracy ». In : *Genome Research* (18 jan. 2017), gr.209601.116. ISSN : 1088-9051, 1549-5469. DOI : 10.1101/gr.209601.116. URL : http://genome.cshlp.org/content/early/2017/01/18/gr.209601.116 (visité le 31/07/2019).

[47] Alexander DOBIN et al. « STAR : ultrafast universal RNA-seq aligner ». In : *Bioinformatics* 29.1 (1er jan. 2013), p. 15-21. ISSN : 1367-4803. DOI : 10.1093/bioinformatics/bts635. URL : https://academic.oup.com/bioinformatics/article/29/1/15/272537 (visité le 21/02/2019).

[48] Manolis MARAGKAKIS et al. « CLIPSeqTools—a novel bioinformatics CLIP-seq analysis suite ». In : *RNA* 22.1 (jan. 2016), p. 1-9. ISSN : 1355-8382. DOI : 10.1261/rna.052167.115. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4691824/ (visité le 31/07/2019).

[49] Philip J. UREN et al. « Site identification in high-throughput RNA–protein interaction data ». In : *Bioinformatics* 28.23 (1er déc. 2012), p. 3013-3020. ISSN : 1367-4803. DOI : 10.1093/bioinformatics/bts569. URL : https://academic.oup.com/bioinformatics/article/28/23/3013/192453 (visité le 31/07/2019).

[50] Kate Voss, Jeff Gentry et Geraldine Van der Auwera. « Full-stack genomics pipelining with GATK4 + WDL + Cromwell ». In : *F1000Research* 6 (8 août 2017). doi : 10.7490/f1000research.1114631.1. url : https://f1000research.com/posters/6-1379 (visité le 26/07/2019).

[51] Simon Anders, Paul Theodor Pyl et Wolfgang Huber. « HTSeq - A Python framework to work with high-throughput sequencing data ». In : *bioRxiv* (19 août 2014), p. 002824. doi : 10.1101/002824. url : https://www.biorxiv.org/content/10.1101/002824v2 (visité le 07/02/2019).

[52] Dongmei Wang et al. « Quantitative functions of Argonaute proteins in mammalian development ». In : *Genes & Development* 26.7 (4 jan. 2012), p. 693-704. issn : 0890-9369, 1549-5477. doi : 10.1101/gad.182758.111. url : http://genesdev.cshlp.org/content/26/7/693 (visité le 29/03/2019).

[53] Aki Fujioka et al. « Dynamics of the Ras/ERK MAPK Cascade as Monitored by Fluorescent Probes ». In : *Journal of Biological Chemistry* 281.13 (31 mar. 2006), p. 8917-8926. issn : 0021-9258, 1083-351X. doi : 10.1074/jbc.M509344200. url : http://www.jbc.org/content/281/13/8917 (visité le 25/03/2019).

[54] Kevin C. Miranda et al. « A Pattern-Based Method for the Identification of MicroRNA Binding Sites and Their Corresponding Heteroduplexes ». In : *Cell* 126.6 (22 sept. 2006), p. 1203-1217. issn : 0092-8674, 1097-4172. doi : 10.1016/j.cell.2006.07.031. url : https://www.cell.com/cell/abstract/S0092-8674(06)01099-3 (visité le 20/08/2018).

[55] Shankar Mukherji et al. « MicroRNAs can generate thresholds in target gene expression ». In : *Nature Genetics* 43.9 (sept. 2011), p. 854-859. issn : 1546-1718. doi : 10.1038/ng.905. url : https://www.nature.com/articles/ng.905 (visité le 14/07/2018).

[56] Jidong Liu et al. « MicroRNA-dependent localization of targeted mRNAs to mammalian P-bodies ». In : *Nature cell biology* 7.7 (juil. 2005), p. 719-723. issn : 1465-7392. doi : 10.1038/ncb1274. url : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1855297/ (visité le 04/09/2020).

[57] Y. H. Wang. « On the Number of Successes in Independant Trials ». In : *Statistica Sinica* 3 (1993), p. 295-312. url : http://www3.stat.sinica.edu.tw/statistica/oldpdf/A3n23.pdf (visité le 09/09/2017).

[58] B. K. Shah. « On the distribution of the sum of independent integer valued random variables ». In : *The American Statistician* 27.3 (1973), p. 123-124. issn : 0003-1305. url : https://www.jstor.org/stable/2683639 (visité le 20/07/2019).

[59] Mika J. Straka. *Poisson Binomial Probability Distribution for Python : tsakim/poibin*. original-date : 2016-05-02T08 :40 :41Z. 16 juin 2019. URL : https://github.com/tsakim/poibin (visité le 20/07/2019).

[60] Toby Mathieson et al. « Systematic analysis of protein turnover in primary cells ». In : *Nature Communications* 9.1 (15 fév. 2018), p. 689. ISSN : 2041-1723. DOI : 10.1038/s41467-018-03106-1. URL : https://www.nature.com/articles/s41467-018-03106-1 (visité le 22/04/2019).

[61] The ENCODE Project Consortium. « An integrated encyclopedia of DNA elements in the human genome ». In : *Nature* 489.7414 (sept. 2012), p. 57-74. ISSN : 1476-4687. DOI : 10.1038/nature11247. URL : https://www.nature.com/articles/nature11247 (visité le 23/04/2019).

[62] Lincoln Stein. *GFF3*. Version 1.24. original-date : 2016-05-02T20 :18 :24Z. 2013. URL : https://github.com/The-Sequence-Ontology/Specifications (visité le 28/07/2019).

[63] Robert Buels et al. « JBrowse : a dynamic web platform for genome visualization and analysis ». In : *Genome Biology* 17.1 (12 avr. 2016), p. 66. ISSN : 1474-760X. DOI : 10.1186/s13059-016-0924-1. URL : https://doi.org/10.1186/s13059-016-0924-1 (visité le 26/06/2019).

[64] John Butcher. *Numerical Methods for Ordinary Differential Equations, 3rd Edition*. 3e éd. Wiley, 2016. URL : https://www.wiley.com/en-us/Numerical+Methods+for+Ordinary+Differential+Equations%2C+3rd+Edition-p-9781119121503 (visité le 01/08/2019).

[65] William H. Press et Saul A. Teukolsky. « Adaptive Stepsize Runge-Kutta Integration ». In : *Computers in Physics* 6.2 (1992), p. 188. ISSN : 08941866. DOI : 10.1063/1.4823060. URL : http://scitation.aip.org/content/aip/journal/cip/6/2/10.1063/1.4823060 (visité le 03/09/2019).

[66] P. Bogacki et L. F. Shampine. « A 3(2) pair of Runge - Kutta formulas ». In : *Applied Mathematics Letters* 2.4 (1er jan. 1989), p. 321-325. ISSN : 0893-9659. DOI : 10.1016/0893-9659(89)90079-7. URL : http://www.sciencedirect.com/science/article/pii/0893965989900797 (visité le 03/09/2019).

[67] C. Runge. « Ueber die numerische Aufl sung von Differentialgleichungen ». In : (1er juin 1895). DOI : 10.1007/bf01446807. URL : https://zenodo.org/record/2178704 (visité le 03/09/2019).

[68] E. Fehlberg. *Low-order classical Runge-Kutta formulas with stepsize control and their application to some heat transfer problems*. 1er juil. 1969. URL : https://ntrs.nasa.gov/search.jsp?R=19690021375 (visité le 03/09/2019).

[69] J. R. Cash et Alan H. Karp. « A variable order Runge-Kutta method for initial value problems with rapidly varying right-hand sides ». In : *ACM Transactions on Mathematical Software* 16.3 (1er sept. 1990), p. 201-222. ISSN : 00983500. DOI : 10.1145/79505.79507. URL : http://portal.acm.org/citation.cfm?doid=79505.79507 (visité le 03/09/2019).

[70] Xiaoye S. Li. « An Overview of SuperLU : Algorithms, Implementation, and User Interface ». In : *ACM Trans. Math. Softw.* 31.3 (sept. 2005), p. 302-325. ISSN : 0098-3500. DOI : 10.1145/1089014.1089017. URL : http://doi.acm.org/10.1145/1089014.1089017 (visité le 01/08/2019).

[71] Timothy A. Davis. « Algorithm 832 : UMFPACK V4.3—an Unsymmetric-pattern Multifrontal Method ». In : *ACM Trans. Math. Softw.* 30.2 (juin 2004), p. 196-199. ISSN : 0098-3500. DOI : 10.1145/992200.992206. URL : http://doi.acm.org/10.1145/992200.992206 (visité le 01/08/2019).

[72] E. Anderson, éd. *LAPACK users' guide*. 3rd ed. Software, environments, tools. Philadelphia : Society for Industrial et Applied Mathematics, 1999. 407 p. ISBN : 978-0-89871-447-0.

[73] *GObject Introspection*. URL : https://gitlab.gnome.org/GNOME/gobject-introspection (visité le 26/07/2019).

[74] Mateo Friggo et Steven G. Johnson. *FFTW*. Version 3.3.7. MIT, 29 oct. 2017. URL : http://www.fftw.org/ (visité le 10/02/2018).