MODÈLES COMBINATOIRES POUR L'ANALYSE DE STRUCTURES D'ARN

Hélène Touzet

Résumé. **—** Le but de cet exposé est de montrer comment des questions de biologie moléculaire peuvent faire émerger de nouveaux modèles combinatoires et de nouveaux algorithmes. Nous présentons plus particulièrement le problème de la comparaison de structures d'ARN, qui fait appel à des représentations sous forme d'arbres ordonnés ou de graphes particuliers, appelés séquences arc-annotées.

1. L'ARN

Chez les organismes vivants, les mécanismes de la cellule sont orchestrés par trois grands types de molécules : l'ADN (acide désoxyribonucléique), l'ARN (acide ribonucléique) et les protéines. Schématiquement, l'ADN porte l'information génétique transmise au fil des générations cellulaires, et les protéines expriment cette information à partir de la transcription et de la traduction des gènes. Dans ce scénario, le rôle des molécules d'ARN est multiple et diversifié. L'ARN intervient dans la machinerie de la synthèse protéique. Il existe également de nombreux petits ARN qui ont des fonctions catalytiques ou qui participent à la régulation de l'expression des gènes.

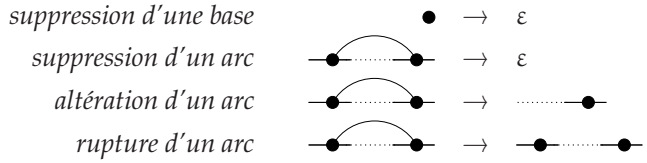
À l'instar de l'ADN, l'ARN est un acide nucléique composé de l'enchaînement de quatre bases, les nucléotides A,C, G et U. Sa spécificité est que c'est une molécule simple brin, qui s'organise de manière hiérarchique en une structure tri-dimensionnelle stabilisée par des appariements entre nucléotides. Cela conduit à une configuration spatiale qui conditionne la fonction de la molécule. La capacité des ARN à intervenir dans des processus métaboliques variés est liée à ces facultés de repliement.

2. Modèles d'édition pour l'ARN

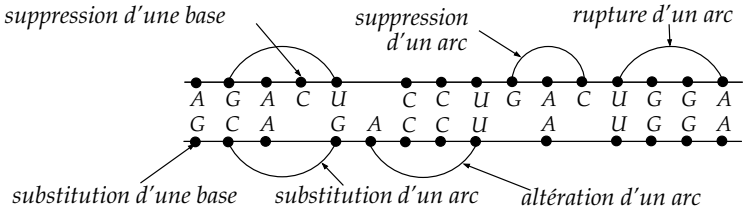
Un des premiers objets de la bio-informatique est de proposer des algorithmes de comparaison. Comparer des molécules permet de les classer en famille, d'identifier les régularités ou les conservations exceptionnelles, d'inférer la fonction par analogie, de retracer le fil de l'évolution. L'ARN n'échappe pas à cette règle. Dans ce cas, la définition d'algorithmes de comparaison passe par le choix d'une représentation combinatoire qui rende compte à la fois de la séquence nucléique et des appariements qui se forment entre les différentes positions. Le modèle le plus expressif est celui des séquences arc-annotées, introduit dans [**5**].

Définition 2.1. — Soit Σ un alphabet fini. Une *séquence arc-annotée* sur Σ^* est définie par un couple (S, P) où S est un mot sur Σ^*, et P un ensemble d'arcs, c'est-à-dire un ensemble de couples de positions de S.

A partir des séquences arc-annotées, il est possible de définir une ou plusieurs instances en fonction de la complexité des interactions autorisées. Nous nous concentrons ici sur le type le plus simple, qui reste néanmoins biologiquement pertinent, celui où les arcs ne forment ni croisements, ni chevauchements. Le choix des opérations d'édition se déclinent également en de multiples modèles évolutifs. Les opérations de *substitution* permettent le renommage d'un nucléotide, que ce nucléotide soit impliqué dans un appariement ou pas. Aux substitutions, il faut ajouter les *suppressions* qui peuvent porter sur un nucléotide ou sur un arc, et sont donc de quatre types.



Ce schéma illustre ces différentes opérations sur un couple de séquences.



À partir de ces opérations d'édition, on définit trois modèles d'expressivité croissante, le modèle III étant le plus fidèle à la réalité biologique.

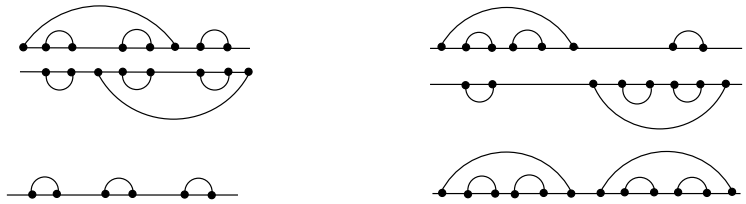
I : opérations de substitution, de suppression de bases et d'arcs (arbres ordonnés)
II : I + altération d'arcs (LAPCS, [**5**])
III : II + rupture d'arcs (edition generale, [**8**])

Définition 2.2. — Soient u et v deux séquences arcs-annotées, et soit K un schéma d'édition (I,II ou III). u est une K-*sous-séquence* de v si u peut être déduit de v par une succession d'opérations de substitution et d'opérations de suppression appartenant au schéma K. Si on associe un poids à chaque opération d'édition, le *coût* de u pour v est la somme des poids de ces opérations. De manière symétrique, v est une K-*super-séquence* de u, de même coût.

Cette définition conduit à deux approches distinctes pour comparer deux séquences arc-annotées, suivant que l'on raisonne en termes de sous-séquence ou de super-séquence :

- chercher la sous-séquence commune de coût optimal ;
- chercher la super-séquence commune de coût optimal.

Ces deux points de vue sont bien sûr équivalents quand on travaille sur des séquences libres d'annotations, sans arcs. Mais ce n'est plus le cas quand interviennent des appariements. L'exemple ci-dessous en témoigne. Il montre un couple de séquences arc-annotées pour lequel la sous-séquence optimale et la super-séquence optimale induisent des scénarios évolutifs distincts.



sous-séquence commune super-séquence commune

3. Résultats de complexité

Les résultats de complexité pour les différents modèles de comparaison sont connus.

	I	II	III
Sous-séquence	$O(n^3 \log(n))$ [**10**]	NP-complet [**11**]	NP-complet [**2**]
Super-séquence	$O(n^4)$ [**9**]	$O(n^4)$ [**3**]	$O(n^4)$ [**3, 6**]

Ce tableau appelle plusieurs commentaires. Il apparait tout d'abord que le choix entre l'approche par sous-séquence et par super-séquence n'est pas anodin en termes de complexité. Dans le premier cas, le problème est rédhibitoire dès que le modèle d'évolution est suffisamment expressif, alors que l'approche par super-séquence permet de considérer toutes les opérations d'édition. Les différents algorithmes polynomiaux procèdent par programmation dynamique, comme la distance d'édition de mots. Le problème de la recherche d'une sous-séquence commune pour le modèle d'édition I n'est autre que le problème de la distance d'édition entre arbres ordonnés. Dans ce cas, les meilleurs algorithmes utilisent des propriétés non-intuitives basées sur la disymétrie des arbres à comparer. C'est ce qui est mis en œuvre dans [**10, 4**].

4. Comparaison d'ARN proches

Si l'on revient à la motivation initiale, la comparaison d'ARN, on se rappelle qu'il s'agit d'analyser des molécules qui partagent une fonction biologique, dérivant d'un ancêtre commun. Dans cette perspective, il est possible d'améliorer les algorithmes de la section précédente en prenant en compte le fait que la distance évolutive, et donc le nombre d'erreurs, est faible. Le nombre d'erreurs s'entend ici comme le nombre de suppressions, qui sont les événements évolutifs qui tendent à modifier la structure de la molécule. Borner le nombre d'erreurs permet, peu ou prou, de restreindre l'espace de recherche des algorithmes de programmation dynamique et donne lieu à des algorithmes exacts plus efficaces.

	I	II	III
Super-séquence avec k erreurs	$O(n \log(n) d^3 k^2)$ [**7**] d, degré des arbres	inconnu	inconnu
Sous-séquence avec k erreurs	$O(k^3 n)$ [**12**]	$O(3,31^k n)$ [**1**]	inconnu

Références

[1] J. Alber, J. Gramm, J. Guo et R. Niedermeier, *Towards Optimally Solving the Longest Common Subsequence Problem for Sequences with Nested Arc Annotations in Linear Time* LNCS 2373, p. 99 – 114, 2002

[2] G. Blin, G. Fertin, I. Rusu et C. Sinoquet, *RNA sequences and the EDIT(NESTED, NESTED) problem*, rapport technique - LINA, Université de Nantes, 2003

[3] G. Blin et H. Touzet, *How to compare arc-annotated sequences : the alignment hierarchy* a paraitre dans les actes de SPIRE 2006, LNCS

[4] S. Dulucq et H. Touzet, *Decomposition algorithms for the tree edit distance problem*, Journal of Discrete Algorithms, 3(2-4), 2005, p. 448-471

[5] P. Evans, *Algorithms and Complexity for Annotated Sequences Analysis*, PhD thesis, University of Victoria, 1999

[6] C. Herrbach, A. Denise, S. Dulucq et H. Touzet, *A polynomial algorithm for comparing RNA secondary structures using a full set of operations.* rapport technique LRI – Université Paris Sud, 2006

[7] J. Jansson et A. Lingas, *A fast algorithm for optimal alignment between similar ordered trees*, Fundamenta Informaticae 56-1(2), p. 105-120, 2003

[8] T. Jiang, G. Lin, B. Ma et K. Zhang, *A general edit distance between RNA structures*, Journal of Computational Biology 9(2), p. 371-388, 2002

[9] T. Jiang, L. Wang et K. Zhang, *Alignment of trees - an alternative to tree edit*, Theoretical Computer Science, 143(1), p. 137-148, 1995

[10] P. Klein, *Computing the edit-distance between unrooted ordered trees*, 1998, p. 91-102, 6th European Symposium on Algorithms

[11] G. Lin, Z.-Z. Chen, T. jiang et J. Wen, *The longest common subsequence problem for sequences with nested arc annotations*, J. of Computer and System Sciences, 65, 2002, p. 465-480

[12] H. Touzet, *A linear tree edit distance algorithm for similar ordered trees*, CPM'05, Lecture Notes in Computer Science, 3537, p. 334-345, 2005

Hélène Touzet
LIFL - UMR CNRS 8022 - Université Lille 1.
E-mail : `Helene.Touzet@lifl.fr`