Recherche d'ARN structurés dans les génomes

Introduction

Outre les approches *ensemblistes* (Boltzmann) vues en cours, le petit monde du repliement de l'ARN connaît actuellement une autre révolution, celle des motifs tertiaires. Ceux-ci sont une généralisation du concept de boucles (Modèle de Turner), incluant potentiellement liaisons tertiaires (Non GC/AU/GU ou interactions de faces particulières) et pseudonoeuds (PK), et associés à des conformations tri-dimensionnelles assez contraintes. Leur incorporation dans les modèles du repliement permet donc une incursion dans le domaine de la 3D. En effet, l'identification de ces motifs dans une structure secondaire *augmentée* connue permet de contraindre très fortement les repliements 3D associés. Un tel programme a permis récemment une avancée majeure dans la détermination de structure 3D directement à partir de la séquence [2] pour des petits ARNs.

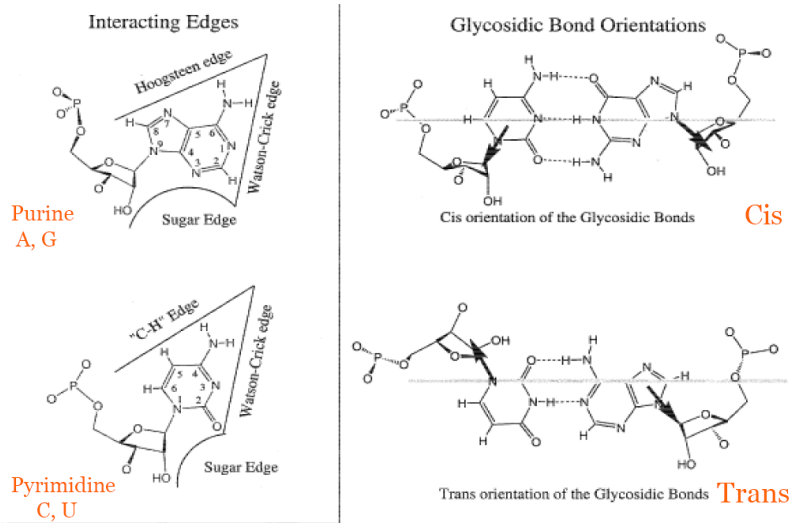


FIGURE 1 – **Gauche :** *Faces* potentiellement impliquées dans des appariements (**W** : Watson-Crick, **S** : Sugar et **H** : Hoogsteen). **Droite :** *Orientations* relatives des bases (**C** : Cis, **T** : Trans).

Les *briques de base* de cette nouvelle vision hiérarchique de la structure d'ARN sont les **appariements non-canoniques**, catégorisés par la nomenclature Leontis/Westhof [1] selon les *faces* impliquées et l'*orientation* relative des brins (Voir Figure 1). Récemment, les appariements de la nomenclature L/W ont été systématiquement analysés selon leur isostéricité – ou similarité géometrique – par superposition exhaustive [3], permettant une quantification de l'impact d'une mutation sur la structure 3D.

Travail demandé

Dans ce projet, nous nous intéressons à l'un des problèmes inverse du repliement, qui consiste à chercher dans un génome un ARN de structures secondaire (augmentée) connue. Les données en entrée de ce problème sont :
– D : Une (longue) séquence D d'ADN

– S : Une structure secondaire *augmentée* (Voir définition ci-dessous, et description du format en Section)

1. Écrire les équations de programmation dynamiques (récurrence) auquel obéit le coût (minimal) de l'alignement optimal sur tout sous-intervalle de la séquence d'ADN D et toute portion de la structure secondaire étendue S (Voir décomposition Section). Ces équations pourront faire l'objet d'une validation par l'examinateur en cours de projet.
2. Implémenter en `python` un programme prenant deux fichiers (Décrivant D et S) en renvoyant l'alignement de coût minimal.
3. Étendre la phase de remontée en proposant la génération de tous les alignements sous-optimaux. Plus précisément, soit k une tolérance et c_0 le coût de la structure optimale, le programme renverra tous les alignements ayant un coût dans l'intervalle $[c_0, c_0 + k]$ (Voir et adapter la technique présentée en cours).

Formalisation

Appariement : Un appariement est entièrement caractérisé par la donnée d'un quintuplet (b_5, b_3, e_5, e_3, o) (cf. Figure 1) où :
– $b5$ et $b3$: Bases en 5' et 3' (**A**, **C**, **G**, ou **U**)
– $e5$ et $e3$: Faces (**W**, **S** ou **H**) en interaction en 5' et 3'
– o : Orientation (**C** : Cis, **T** : Trans)
On notera dans la suite par $\mathcal{A} = \{A, C, G, U\}^2 \times \{W, S, H\}^2 \times \{c, t\}$ l'ensemble des appariements (canoniques ou non-canoniques).
On étend la notation parenthésée en une **notation parenthésée étendue** (NPE) pour la structure secondaire augmentée en :
– Mettant en indice des parenthèses le type précis d'appariement.
– Factorisant chaque région non-appariée (séquence de m bases non-appariées consécutives) en un unique caractère $\bullet_m$.
Par exemple, une hélice simple composée de deux appariements $p, p' \in \mathcal{A}$ encadrant 5 bases non-appariées, dénoté par `((.....))` dans la notation parenthésée classique, donnera lieu à la NPE $(_p\ (_{p'} \bullet_5\ )_{p'})_p$.

Alignement : Un alignement consiste en la donnée, pour chaque position de la structure secondaire (augmentée), d'une position correspondante dans la séquence d'ADN, ou `-1` si celle ci n'est associée à aucune position (Délétion).

Exemple

```
              200       210            220       230       240
               |         |              |         |         |
ADN       ...GCGGCCAATGTGGAGGTCG----ATGATCGATCTCATTATGGCCGC...
               |||||||  ||||||||||    ||||||||||||  |||||||||
SecStr         (((((((--...(((((((.........)))))))--..)))))))
               |           |         |         |           |
               0           10        20        30          40
```

Par exemple, l'alignement ci-dessus (Notation classique utilisée pour la structure secondaire) sera représenté en `python` par la liste :

```
[(0,200),(1,201),...,(5,205),(6,206),
 (7,209),(8,210),...,(15,217),(16,218),
 (17,-1),(18,-1),(19,-1),(20,-1),
 (24,219),(25,220),...,(31,229),(32,230),
 (33,233),(34,234),...,(40,240),(41,241)]
```

Coût d'un alignement : Le coût à minimiser vise à favoriser la conservation de la structure 3D. Celle-ci induit alors les contraintes suivantes :

A. Les mutations sont autorisées dans les régions appariées. On pénalise de telles mutations proportionnellement au changement conformationnel induit.

B. Les mutations sont autorisées sans pénalité dans les régions non-appariées.

C. Les insertions/délétions de bases sont autorisées, mais pénalisées, dans les régions non-appariées. Une *insertion* correspond à une portion de l'ADN sans équivalent dans la structure d'ARN (Ex. : Bases 207–208 et 231–232 dans l'alignement ci-dessus), et une *délétion* à une portion de la structure d'ARN sans équivalent dans la séquence d'ADN (Ex. : Bases 17–20).

Cela se traduit numériquement par l'introduction d'une **fonction de coût**, additive sur les différents éléments d'un alignement, et définie sur ces éléments atomiques comme suit :

A. **Mutation/Conservation d'appariement :** Celles ci se voient associer un coût individuel, et on supposera disponible une fonction $\delta : \mathcal{A} \times \{A, C, G, T\}^2 \to \mathbb{R}$ qui prend en paramètre un appariement $a \in \mathcal{A}$ et un couple $(b_5, b_3) \in \{A, C, G, T\}^2$ de bases ADN, et renvoie un score lié à la modification géométrique induite par le changement de bases dans l'appariement.

B. **Mutation de base non-appariée :** Non pénalisées, elles ont un coût nul (0).

C. **Insertions/délétions de base non-appariée :** Le coût, fixe, d'une insertion (resp. délétion) est de λ (resp. α).

Remarque 1 : Dans les régions appariées, le coût d'une substitution est plus faible (nul) que la somme d'une insertion/délétion. Comme, par ailleurs, tous ces coûts sont indépendants de leur position au sein d'une région non-appariée, alors **on impose** [1] **que les insertions et délétions soient regroupées au début de la zone non-appariée**. On évite ainsi une explosion combinatoire d'alignements équivalents, phénomène problématique pour la génération d'alignements sous-optimaux *réellement* différents.

Exemples

```
D     GGAUACC          AUGGAUACCAA          AAAUGGAUACCAAUG
      |||||||            |||||||              |||||||
S     ((...))          --((...))--          ----((...))----
```

Remarque 2 : On s'intéresse ici à un **alignement local** de la structure secondaire dans l'ADN. En conséquence, on ne comptera pas de pénalité pour les séquences insertions préfixe et suffixe de la structure secondaire. En conséquence, les trois alignements ci-dessus se verront associer le même coût.

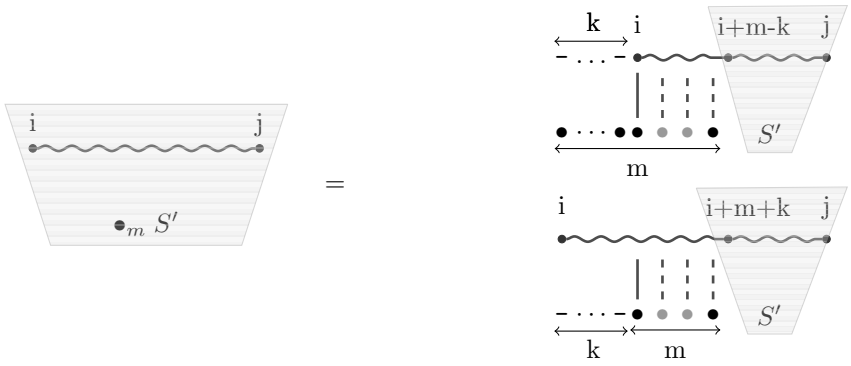
1. Sans perte de généralité pour la partie minimisation du coût ...

Décomposition

Nous allons nous attacher à construire une décomposition pour l'ensemble des alignements d'un sous-intervalle (i, j) de la séquence D d'ADN et S la structure d'ARN. L'écriture des équations de programmation dynamique **fait partie du travail demandé**.
En se concentrant sur la première position de la structure secondaire d'ARN, on remarque immédiatement que seuls deux cas apparaissent :

– $S = (_p S')_p S''$: L'interdiction d'insérer des éléments dans les hélices implique que la première base i est appariée avec un certain $k, i < k \leq j$, dans l'ADN. Cet appariement induit un alignement de S' (resp S'') avec le sous-intervalle $(i+1, k-1)$ (resp. $(k+1, j)$).

– $S = \bullet_m S'$: Conformément à la remarque ci-dessus, l'alignement d'une région non-appariée donne lieu à une séquence d'inclusions **ou** de délétions, suivie de l'alignement (Mutation ou conservation) des bases restantes avec une portion de taille égale dans l'ADN. Une fois ceci fait, on recherche S' dans l'intervalle restant de l'ADN.



Enfin, on va distinguer un *ancrage initial* de la structure secondaire recherchée dans un sous-intervalle de la séquence d'ADN. En effet, on doit éviter de compter dans la fonction de coût les insertions préfixe et suffixe de l'alignement (Voir Remarque 2 ci-dessus).

Formats d'entrée

Pour la séquence d'ADN D, le ficher sera uniquement constitué de la séquence. On pourra aussi (non-obligatoire) accepter le format FASTA, qui fragmente la séquence sur plusieurs lignes, et autorise l'insertion d'informations supplémentaires (à ignorer ici).

Pour la structure secondaire S, le programme devra accepter en entrée deux types de fichiers :

A. Les notations parenthésées vues en TP.

Exemple
```
((((.((((....)))).((((....)))).((((....)))).))))
GGCGAGUCGUAGCCGACUAGGCUAGUGCCUUGUACUAUGGUGCUCGCC
```

Comme celles-ci ne permettent pas de spécifier les *faces* ni l'*orientation*, on supposera que tous les appariements impliquent les faces Watson/Crick et d'orientation Cis, ce qui donne un encodage (G, C, W, W, c) pour l'appariement le plus extérieur (Pos. 1/48) dans l'exemple ci-dessus.

B. Une version adaptée du format CT (Utilisé par MFold), où chaque ligne contient :
- Position de la base
- Base
- Position de la base appariée, ou 0 si non-appariée
- Si appariée, face du côté 5'
- Si appariée, face du côté 3'
- Si appariée, orientation

Exemple
```
1 G 8 W W c
2 G 7 W H t
3 C 0
4 A 0
5 U 0
6 U 0
7 U 2 W H t
8 C 1 W W c
```

→ Paire G/C Watson/Watson, orientation Cis (Canonique)
→ Paire G/U Watson/Hoogsteen, orientation Trans
→ Base non-appariée
→ ...
→ ...
→ ...
→ ...
→ Paire G/U Watson/Hoogsteen, orientation Trans
→ Paire G/C Watson/Watson, orientation Cis (Canonique)

Remarque : On supposera que les fichiers fournis décrivent des structure sans croisement (Pas de pseudo-noeuds).

Contact

N'hésitez pas à me contacter (yann.ponty[AT]lix.polytechnique.fr) si un point vous semble obscur ...

Références

[1] N. Leontis and E. Westhof, *Geometric nomenclature and classification of RNA base pairs*, RNA **7** (2001), 499–512.

[2] M. Parisien and F. Major, *The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data*, Nature **452** (2008), no. 7183, 51–55.

[3] Jesse Stombaugh, Craig L Zirbel, Eric Westhof, and Neocles B Leontis, *Frequency and isostericity of RNA base pairs.*, Nucleic Acids Res **37** (2009), no. 7, 2294–2312.