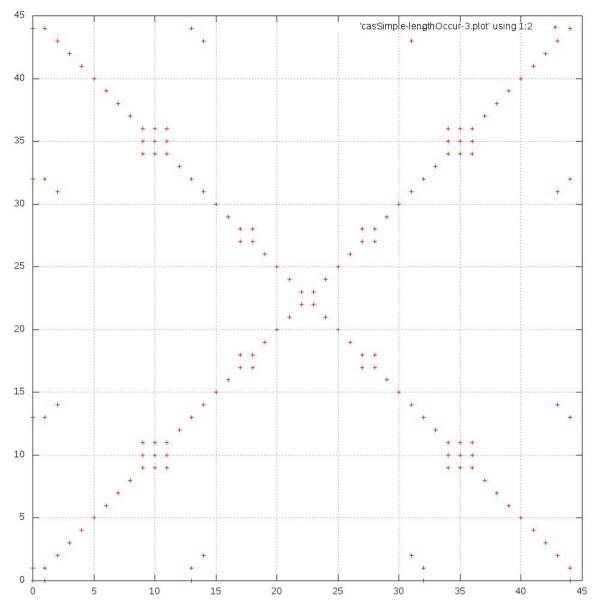
Compte rendu TP2 Recherche de pré-microARN et Hybridation ARNmessager / MicroARN

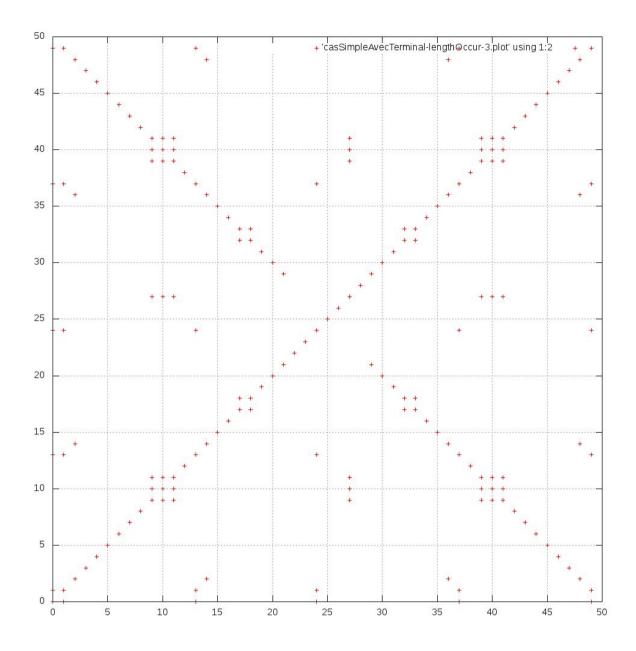
Partie 1

Afin de pouvoir réaliser des appariement de micro ARN à partir de pré-microARN nous avons réalisé quatres exemples (Ces exemples ainsi que les DotPlot associé se trouvent dans le dossier fasta):

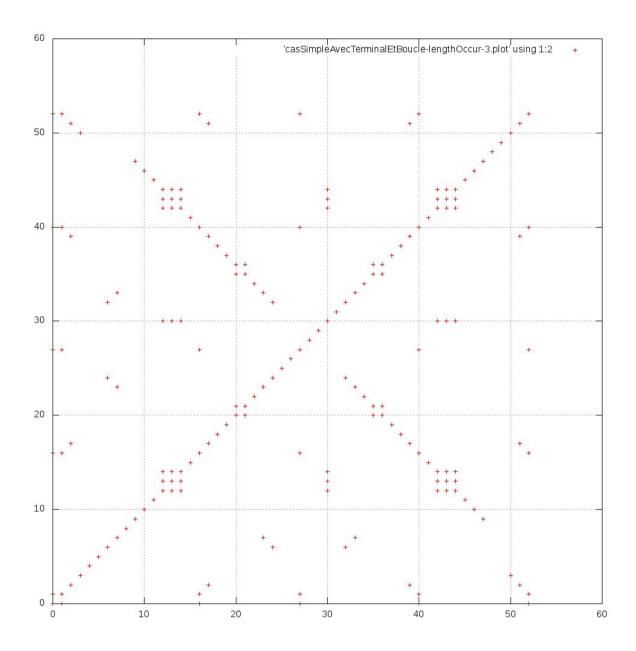
- Un cas simple ne contenant pas de boucle terminale et pas de boucle intermédiaires:



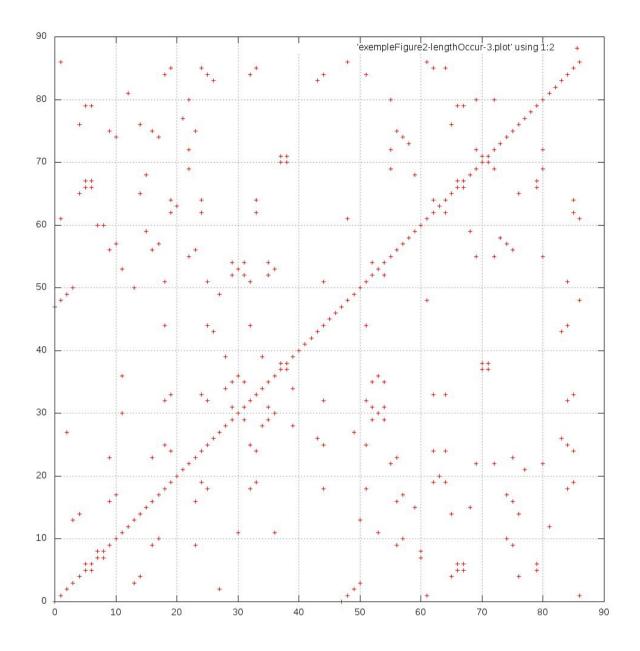
- Un cas simple présentant une boucle terminale:



- Un cas simple présentant une boucle terminale et une boucle intermédiaire:



- Un cas complexe avec terminale et boucles diverses:



Il est possible d'exécuter manuellement la réalisation des dotplot avec la suite de commande suivante:

java -jar {chemin vers l'archive TP1.jar} {chemin vers le fichier fasta} 3 D RC

enfin la suite de commande à exécuter dans gnuplot (exemple pour le fichier casSimple.fasta):

set terminal 'png' size 1000,1000 set grid

set output 'casSimple.png' plot 'casSimple-lengthOccur-3.plot' using 1:2

Partie 2

exempleFigure2.fasta

Afin de réaliser un appariement de nucléotides permettant de valider une séquence micro ARN nous avons utilisé deux méthodes:

- La première consiste à récupérer tous les mots de 3 caractères dans la séquence. En plaçant deux curseurs un au début (curseur i) et un à la fin (curseur j) de la séquence on se déplace d'un caractère à la fois. Ainsi pour chaque mot de 3 caractères pointé par le curseur i on regarde si il existe une occurrence reverse complémentaire entre 3 et 6 caractères avant le curseur j (une zone de gap est d'au maximum 3 caractères). Si le mot est trouvé on déplace i de un caractère et j du nombre de gap à réaliser pour obtenir le mot. Si non on ne déplace uniquement que i.
- La seconde méthode consiste à réaliser un alignement selon la méthode Needleman et Wunsch. Afin de réaliser cette méthodes nous avons réalisé certaines modifications :
 - Les chaînes comparées sont le micro ARN et le reverse du micro ARN
 - La condition de fin de récurrence est **i** == **j** c'est à dire lorsque les deux curseur ont atteint le milieu de la séquence.
 - Le référentiel d'égalité a changé selon le principe de reverse complémentarité
 - Le référentiel d'égalité réalise une égalité sur des mots de 3 caractères.
 - Les substitutions sont interdites

Nous avons décidé d'utiliser cette deuxième méthode qui retourne de meilleurs résultats que la première mais qui est beaucoup plus difficile à mettre en place. Toutefois ces deux algorithme ne fonctionnent que pour les cas simple voici les résultats obtenus pour les quatres exemples:

UGCUUCCGGCCUGUUCCCUGAGACCUCAAGUGUGAGUGUACUAUUGAUGCUUCACA CCUGGGCUCUCCGGGUACCAGGACGGUUUGAGCA

La difficulté de cette deuxième méthode est de mettre en place un algorithme glouton qui doit s'adapter aux contraintes de validation d'appariement. En effet, cas d'égalité entre insertion et délétion il faut pouvoir choisir le cas optimum qui permet de respecter les contraintes de 3 gap successif. Ainsi à cause de ce dernier élément nous n'arrivons pas à réaliser un appariement valide pour le cas complexe.

Il est possible d'exécuter manuellement le programme d'appariement avec la commande suivante:

java -jar {chemin vers Appariement.jar} {chemin vers le microARN}

Enfin un script réalise l'appariement de tous les micro ARN contenu dans un dossier.

./EvaluerAppariement {dossier contenant les fichier micro ARN}

Partie 3

Afin de pouvoir évaluer le taux d'hybridation entre un ARN messager et un micro ARN nous avons décidé de réaliser encore une fois l'algorithme de Needleman et Wunsch en utilisant cette fois comme référentiel d'égalité un caractère unique selon le principe de complémentarité.

La récurrence est réalisé sur toutes les sous séquences de l'ARN messager commençant par une succession de 7 nucléotides complémentaires avec les 7 premiers nucléotides de la séquence de micro ARN. La taille de la sous séquence d'ARN Messager doit être au maximum supérieure à 5 nucléotides par rapport à la séquence de micro ARN (on considérera que si sous séquence contient plus de 5 gap elle n'est pas hybridable).

Enfin, pour chaque sous séquence calculée un score est calculé selon la formule suivante:

2 * nombre de match + nombre de substitutions - nombre de gap

Ainsi si deux sous séquences peuvent être trouvée pour un ARN messager l'alignement obtenant le score le plus élevé sera gardé.

Les résultats obtenus sont tous présents dans le fichier Résultat_Hybridation.pdf Les résultats présentent le taux d'hybridation sous deux formes:

- forme naïve : nombre de match / taille du micro ARN * 100
- forme complexe : score calculé / (taille du micro ARN * 2) * 100

Pour chaque ARN messager les meilleurs cas d'hybridation sont les suivants:

ARN messagé : ARNmessager-1.fasta	
Micro ARN : hsa-miR-19a_MI_ARN.fasta	
Résultat:	
UCAAGAUCCAUCGACCUUAAACAGG	
XX X X XXX	
AGUUUUGCAUAGUUGCACUACA	
Pourcentage d'hybridation méthode naif : 68.10%	
Pourcentage d'hybridation méthode calculée : 72.7%	

ARN messagé : ARNmessager-2.fasta	
Micro ARN : hsa-miR-18a_MI_ARN.fasta	
Résultat:	

GUUCCGCCUGGGC-UGUGGGCAUGUC	
X X XX-X-	
UAAGGUGCAUCUAG-UGCAG-A-UAG	
Pourcentage d'hybridation méthode naif : 73.9%	
Pourcentage d'hybridation méthode calculée : 69.5%	

ARN messagé : ARNmessager-3.fasta		
Micro ARN : hsa-miR-20a_MI_ARN.fasta		
Résultat:		
AUUUUAUGACUAUUUAUGC-ACAUU		
X X- X-X-		
UAAAGUGCUUAUAGU-GCAGG-UAG		
Pourcentage d'hybridation méthode naif : 78.2%		
Pourcentage d'hybridation méthode calculée : 71.7%		

	, , , , , , , , , , , , , , , , , , , ,
ARN messagé : ARNmessager-4.fasta	ARN messagé : ARNmessager-4.fasta
Micro ARN : hsa-miR-17-5p_MI_ARN.fasta	Micro ARN : hsa-miR-20a_MI_ARN.fasta
Résultat:	Résultat:
GUUUUGUGUG-UGUGUGUGU-GUG	GUUUUGUGUG-UGUGUGUGU-GUG
X X- X	X X- X
CAAAGUGCU-UACAG-UGCAG-GUAG	UAAAGUGCU-UAUAG-UGCAG-GUAG
Pourcentage d'hybridation méthode naif : 73.9%	Pourcentage d'hybridation méthode naif : 73.9%
Pourcentage d'hybridation méthode calculée : 63.0%	Pourcentage d'hybridation méthode calculée : 63.0%

Comme l'énonce le paradox de simpson il est difficile dire si deux séquences sont hybridable car il faut définir plusieurs variables qui peuvent ne pas s'associer. Ainsi Nous avons décidé de définir deux séquences hybridables lorsque leur pourcentage naïf et calculé dépasse les 60%.

Il est possible d'exécuter manuellement le programme de calcul d'hybridation en utilisant la commande suivante:

java -jar {chemin ver Hybridation.jar} {chemin vers le fichier ARN messager} {chemin vers le fichier micro ARN}

Enfin il est possible de réaliser une suite de test en mettant dans un dossier les fichiers ARN messager et un autre les micro ARN et lancer la commande suivante:

./EvaluerHybridation.sh {chemin vers le dossier ARN messager} {chemin vers le dossier micro ARN}

Structure de l'archive

Compte_rendu_TP2.pdf Resultat_Hybridation.pdf src (dossier de sources)

TP1.jar (permet de lister tous les mots de n caractère dans une séquence ADN: java -jar TP1 {fichier fasta} 3 D RC)

script-gnuplot.plot (permet de générer les dotplot des exemples présenté partie 1) fasta (dossier contenant les fichier exemples partie 1-2)

arnMessager (contient les ARN messager partie 3)

microArn (contient les micro ARN partie 3)

EvaluerAppariement.sh (réalise l'appariment sur tous les fichiers d'un dossier précisé)

EvaluerHybridation.sh (realiser l'hybridation de tous les fichiers arn messager et micro arn)

Appariement.jar

Hybridation.jar