

**Table des matières**

[**INTRODUCTION**](#_30j0zll) **3**

[**APPROCHE:**](#_zhavzujcqmse) **3**

[**PSEUDO CODE**](#_nmicfybiskzq) **4**

[**RÉSULTATS**](#_4epdp2p10q20) **4**

[**CONCLUSION**](#_vli712v070y3) **4**

[**RÉFÉRENCE**](#_6o1lpg7j5v2m) **5**

### **INTRODUCTION**

L'alignement épissé consiste à aligner les produits géniques épissés sur des séquences d'ADN non épissées. Elle est importante pour l'annotation du génome, la prédiction de gènes, l'identification de structures de gènes et d'autres études d'épissage. La qualité de l’annotation du transcriptome est grandement améliorée avec l’utilisation de Données sur l'ARN des gènes ou génomes apparentés. L’alignement multiple épissé quant à lui consiste à aligner en parallèle plusieurs alignement épissé de séquence d’ARN sur une seul séquence de gène. Cet alignement permet d’améliorer le ratio de la prédiction de la structure des gènes en ciblant simultanément de nombreux ARN.

Ce projet consistait donc à trouver une solution au problème d’alignements multiples épissés de familles de gènes à partir d’alignements par paire. Après notre première rencontre avec Professeur Aïda Ouangraoua, nous avons décidés de nous rencontrer tous les mercredis afin de travailler sur le projet. Lors de nos rencontre, nous nous tenions au courant de l’avancement de chacun et nous travaillions sur les problèmes de l’un et de l’autre si jamais il y en avait sinon nous procédions à l’étape suivante qui était de tout mettre en commun et de s’assurer que nos travaux allaient parfaitement ensemble. Ensuite nous faisons de nouvelles recherche pour améliorer ou avancer notre projet avant de se répartir de nouvelles tâches pour la semaine prochaine.

Nous avons décidés de faire des recherches concernant les alignements épissés afin de comprendre les données que nous avions en entrée avant de passer aux alignements multiples. Nous avons donc compris que le problème d’alignements multiples épissés de familles de gènes est une extension du problème d’alignements par paire épissés. En effet, l’alignement par paire epissé correspond aux entrées que nous avions pour notre projet. Ces alignements sont habituellement utilisés pour la prédiction et/ou l’annotation de génome; pour ce faire on trouve un alignement épissé optimal entre une séquence codante et un gène qui capturera toutes les séquences similaires. Nous pourrons donc utilisés la solution au problème d’alignements épissés pour trouver un alignement multiple épissé optimal entre un groupe de séquences codantes et un groupe de gènes. Il nous faudra donc fusionner progressivement les alignements par paires tout en filtrant les sous-alignements qui pourraient être erronées ainsi on arrivera a transformé les alignements par paires épissés en alignements multiples épissés. Nous détaillerons plus spécifiquement notre approche ci dessous.

### 

### **APPROCHE:**

### 

### Lors de nos recherche nous avons aussi découvert un programme nommé Mirage qui fait aussi des alignements de séquences multiples épices. Le fonctionnement de ce programme nous a un peu aidé dans la compréhension de l’alignement et aussi dans la confection de notre programme.

### 

### En effet, la construction d’alignement multiple épissés avec Mirage se fait en quatre étape en prenant en entrée un fichier de références et une base de données remplies de séquences de protéines qui ont été formatées avec FASTA.

### 1-Les séquences de protéines sont assemblées à leur génome respectifs. FastMap est utilisé pour itérer à travers la bases de données et générer des mappages génomiques épissés et lorsque des similarités sont trouvés Mirage essaye de les mappés ensemble.

### 2-Le mapping de l’étape précédente est ensuite utilisé pour produire des alignements entre les espèces.

### 3-Les alignements sont ensuite fusionnés pour former un alignments multiples épissés

### 4-Dans la dernière étape, les singletons qui n’ont pas été traités sont finalement alignés avec l’alignement multiple.

### 

Nous avons aussi trouvé un article écrit par les professeurs Aïda Ouangraoua, Jean-David Aguilar, et Safa Jammali qui parles du problèmes des alignments épissées et s'étendait aussi sur le problèmes des alignements multiples épissés les données et algorithmes données dans cet article nous a énormément dans le développement de notre solution.

### Ces différentes recherches et à l’aide des détails fournis par Professeur Aïda Ouangraoua nous ont permis de structurer un code qui nous permet de traiter le problème d’alignements multiple épissés. Mais avant observons et notons les entrées.

### 

### Dans le tableau qui suit nous notons les entrées pour nous retrouver plus facilement.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Segment** | **Gene** | **Size** | **Cons %** | **No** |  |  |  |  |
| **ENST00000346333** | **ENSG00000158483** | **396** | **100.00** | **1** |  |  |  |  |
| **Segment** | **Gene** | **Size** | **a** | **b** | **s** | **e** | **Cons** | **Exon** |
| **ENST00000346333** | **ENSG00000158483** | **96** | **0** | **96** | **6027** | **6123** | **1.00** | **TC<Exon>GT** |
| **ENST00000346333** | **ENSG00000158483** | **63** | **96** | **159** | **8273** | **8336** | **1.00** | **AG<Exon>GT** |
| **ENST00000346333** | **ENSG00000158483** | **150** | **159** | **309** | **14507** | **14657** | **1.00** | **AG<Exon>GT** |
| **ENST00000346333** | **ENSG00000158483** | **87** | **309** | **396** | **18051** | **18138** | **1.00** | **AG<Exon>AC** |
| **Segment** | **Gene** | **Size** | **Cons%** | **No** |  |  |  |  |
| **ENST00000359244** | **ENSG00000158483** | **498** | **100.00** | **1** |  |  |  |  |
| **Segment** | **Gene** | **Size** | **a** | **b** | **s** | **e** | **Cons** | **Exon** |
| **ENST00000359244** | **ENSG00000158483** | **96** | **0** | **96** | **6027** | **6123** | **1.00** | **TC<Exon>GT** |
| **ENST00000359244** | **ENSG00000158483** | **63** | **96** | **159** | **8273** | **8336** | **1.00** | **AG<Exon>GT** |
| **ENST00000359244** | **ENSG00000158483** | **102** | **159** | **261** | **11870** | **11972** | **1.00** | **AG<Exon>GT** |
| **ENST00000359244** | **ENSG00000158483** | **150** | **261** | **411** | **14507** | **14657** | **1.00** | **AG<Exon>GT** |
| **ENST00000359244** | **ENSG00000158483** | **87** | **411** | **498** | **18051** | **18138** | **1.00** | **AG<Exon>AC** |

Dans ce tableau size correspond à la taille du segment ARN, a représente le début du segment et b la fin, s représente le début de l’alignement sur le genes et e la fin. Cons % est le pourcentage de conservation (pourcentage de l'identité des nucléotides).

Notre algorithme est constitué de 3 parties:

Notre première partie consiste à aligner les blocs (segments) en multi-blocks.

Pour cela nous avons regardés les compatibilités des blocs entre eux. Pour savoir si deux blocs sont compatibles, nous observons leur alignements sur le gène c'est à dire que nous prenons en compte s et e. On désigne ensuite une variable t à la quel on attribuera la valeur 50 qui nous servira de seuil.

Ainsi:

Pour tous les blocs qui sont alignés sur le gène, on les aligne deux par deux de cette façon.

Si s1 = s2 ou e1 = e2 +/- t

Ou Si s1 = s2 +/- t ou e1 = e2

Alors ils sont compatibles on crée donc un nouveau noeud avec qui est identifier par l’id du Gene, la taille s et la fin e correspondant et on rajoute la sequences aligné sur cette partie du gène comme étant “enfant” du noeud.

Ensuite nous mettons tous les noeuds dans un tableau et on cherche une compatibilités entre chacun des noeuds dans ce cas la on regarde encore une fois s et e. Nous avons un seuil de 15 cette fois ci. Nous observons si

s1 = s2 ou e1 = e2 +/- 15

s1 = s2 +/- 15 ou e1 = e2

Si c’est le cas alors les noeuds sont compatibles et on les même dans le même bloc de compatibilité. Et grâces à notre première étape on arrive à se rendre compte que chacun des noeuds compatibles ont les même séquences comme “enfants” avec les mêmes nom de séquences, même a, même b.

### **PROBLÈME RENCONTRÉ :**

Par Contre avec notre méthode il arrive que certains noeuds aient les mêmes enfants mais ne soient pas compatibles on aurait aimé qu’ils soient compatibles et soit mis ensemble et considérer comme faisant partie de l’alignement multiple.

Ce problème arrive car ces noeuds qui ont les mêmes enfants ne satisfont pas les contraintes de compatibilités. En effet un des noeuds peut avoir s1= 14873 et l’autre s2 =18093 avec e1 = 15704 et e2 = 18402. Notre seuil dans ce cas la n’est pas suffisamment grand pour les apparier et les considérées comme compatible.

Nous n’avons pas eu le temps de l'implémenter mais l’idée que nous avons eu était de récupérer ces noeuds et d’essayer de les rendre compatibles avec les blocs compatibles déjà former. Pour cela nous n’allions pas passer par notre formule précédentes mais nous allions plutôt regarder les enfants de chaque noeuds et les comparer avec les enfants des noeuds qui se trouvent déjà dans des blocs de compatibilité. Si jamais ils ont les mêmes enfants alors on rajoute le noeuds au blocs de compatibilité. Si non le noeud restera incompatible avec les autres noeuds .

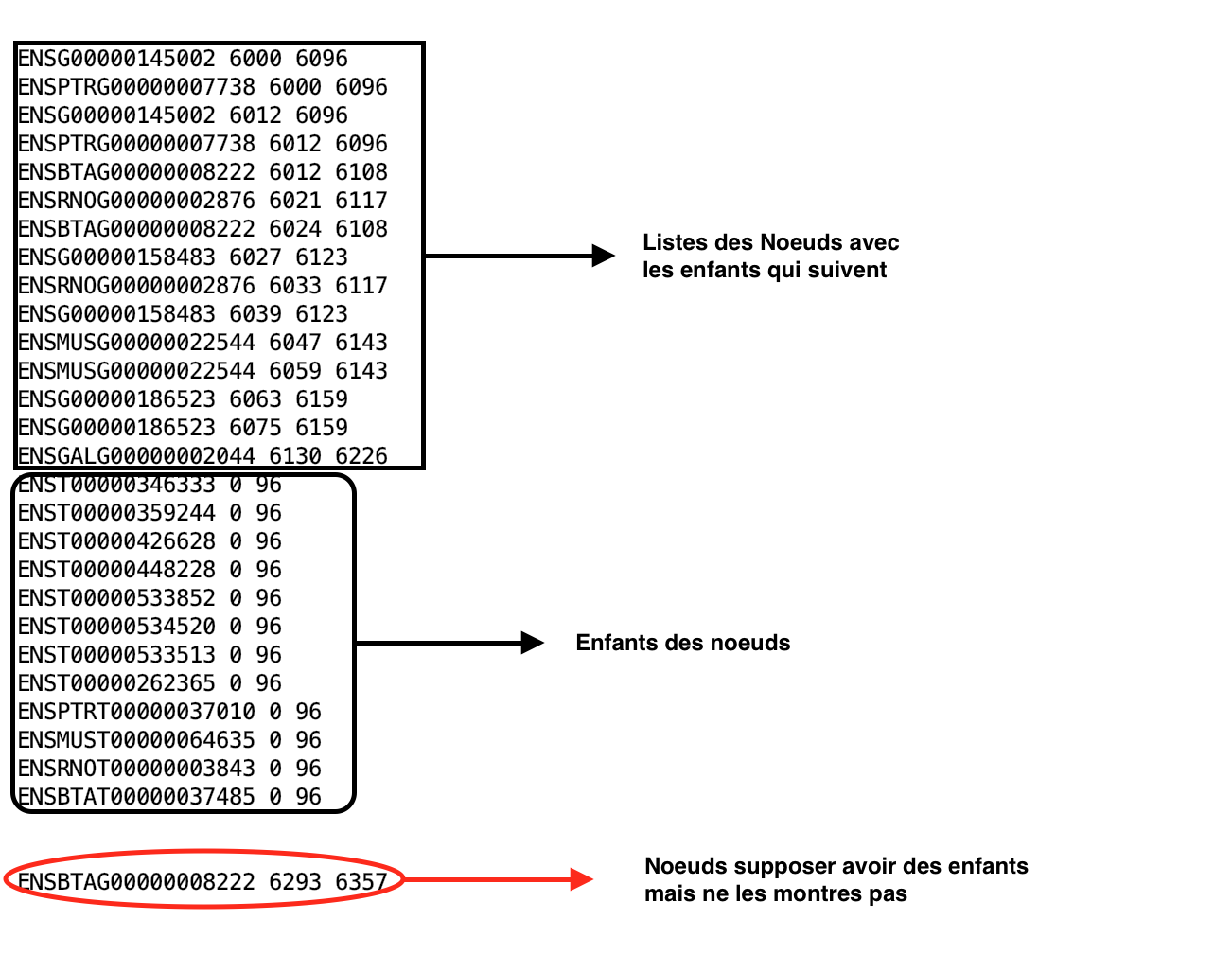
N'étant pas très expérimenté en python nous avions pris une approche au début de notre projet qui nous a causé des problèmes surtout au niveau de la lecture du fichier nous avions énormément de données qui se réitérait et avons passez beaucoup de temps à vouloir résoudre le problème et avons ensuite décidé de le lire avec la structure que nous avons actuellement. Une structure orienté objets qui a des classes représentants les gènes, les segments, les alignements, les noeuds, et les blocs que nous créons/ lisons.

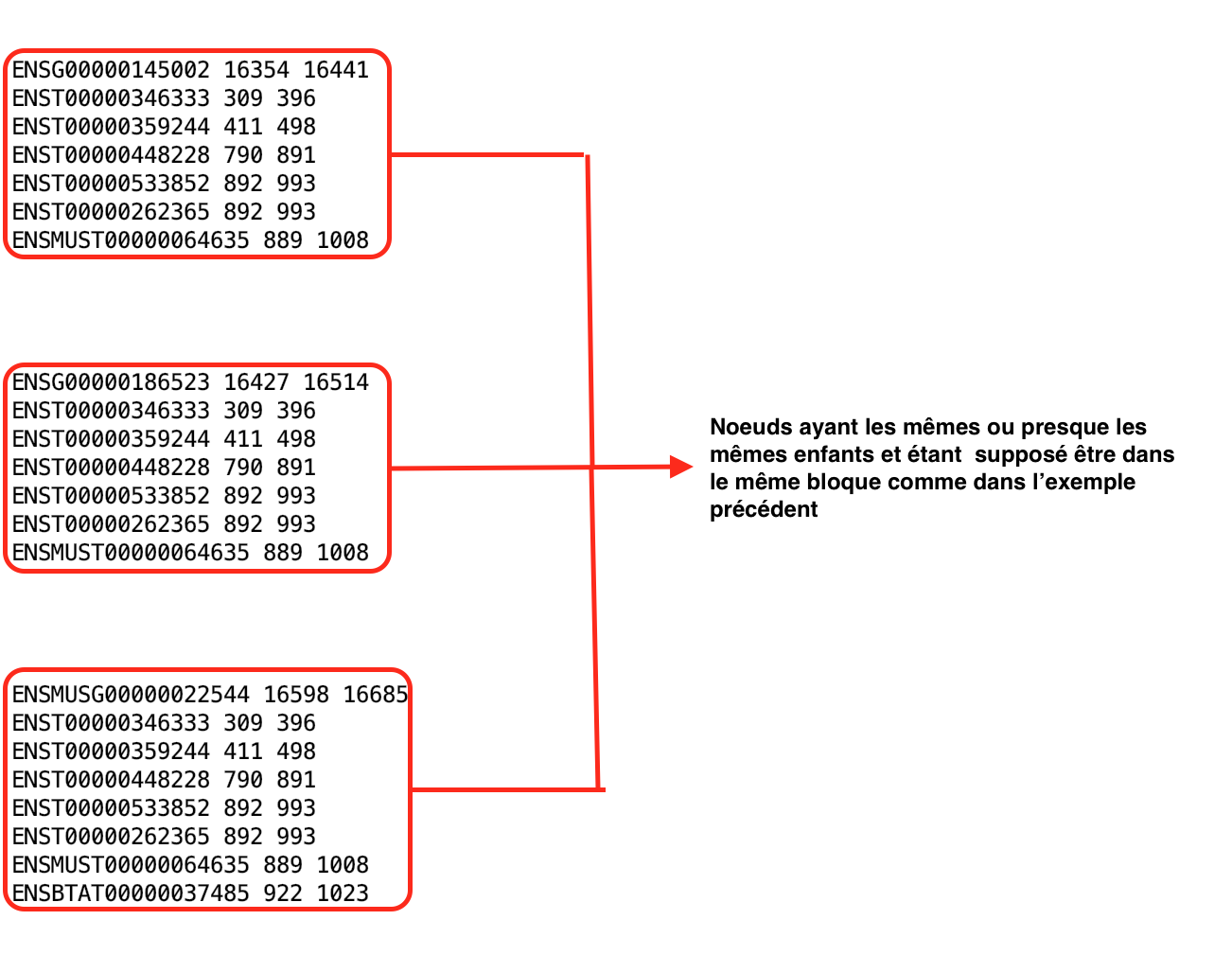
Un autre soucis auxquels nous avons fait fasse était l'intégration et l’utilisation de la bibliothèque été à notre programme nous n’avions pas bien compris son intérêt et son fonctionnement vis à vis de notre projet et avons décidé de programmer sans la bibliothèque.

Dans nos sorties certains Noeuds ou groupe de Noeuds apparaissent mais la liste des “enfants” de ce/ces Noeud(s) n’apparaissent pas car ne sont étrangement pas pris en compte.

### **RÉSULTATS**

Nos resultats sont presentes de la manière suivante:

****

****

### **CONCLUSION**

En conclusion ce projet nous a aidé à comprendre comment on pouvait approcher le problèmes des alignements multiples épissés et comment une bonne résolution du problèmes des alignements multiples épissés pouvait nous aider à bien repérer les similarités entre les différents gènes et en se basant sur les séquences qui les codes.

### 

### 

### **RÉFÉRENCE**

[Improving spliced alignment for identification of ortholog groups and multiple CDS alignment](https://arxiv.org/pdf/1709.06169.pdf)

<https://www.researchgate.net/publication/327213392_Splice-Aware_Multiple_Sequence_Alignment_of_Protein_Isoforms>