

**Rapport de Stage M2**

*Goulancourt Rebecca*

*M2 Biologie-Informatique*

*Université de Paris*

*2021-2022*

*Github : https://github.com/Rebbekkah/Stage\_M2.git*

*Tutrice : Ingrid Lafontaine, IBPC.*

*Mots clés : Chloroplaste, protéine, adressage, régulation post-transcriptionelle, alpha-solénoïde, prédiction, modèle de prédiction « Random Forest ».*

*SOMMAIRE*

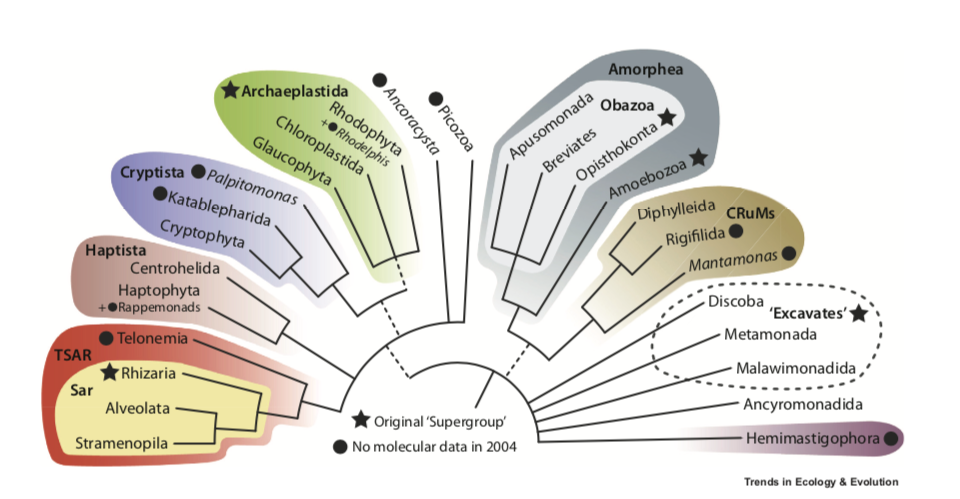
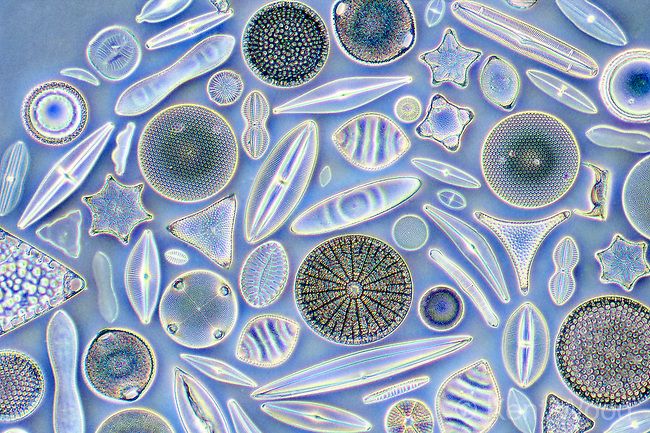
1. Introduction & état de l’art

Les diatomées sont des organismes marins, microalgues eucaryotes photosynthétiques de diverses formes. C’est-à-dire qu’elles possèdent différents axes de symétrie. Leur taille peut varier de 2 µm à 1mm et en tant qu’espèce photosynthétique ils sont dotés de mitochondrie et surtout de chloroplastes. Ce chloroplaste leur vient d’une endosymbiose secondaire. L’endosymbiose secondaire est caractérisée par l’internalisation d’une cellule issue d’une endosymbiose primaire. Les cellules issues de l’endosymbiose primaire, comme les algues vertes, proviennent de la phagytose d’une cyanobactérie par une cellule ancestrale primaire hétérotrophe. La cyanobactérie se différencie ensuite en chloroplaste et la cellule acquiert une capacité photosynthétique, devenant ainsi autotrophe. Cependant les diatomées faisant partis de la famille des *Stramenopila* (algues rouges) découlent de l’internalisation d’une cellule autotrophe possédant un chloroplaste par une autre cellule hétérotrophe, obtenant ainsi des caractéristiques photosynthétiques. Le chloroplaste, au même titre que la mitochondrie, est un organite. Par définition il possède donc du matériel génétique qui s’exprime et lui est propre.

A

B

Algues vertes



Endosymbiose 1aire

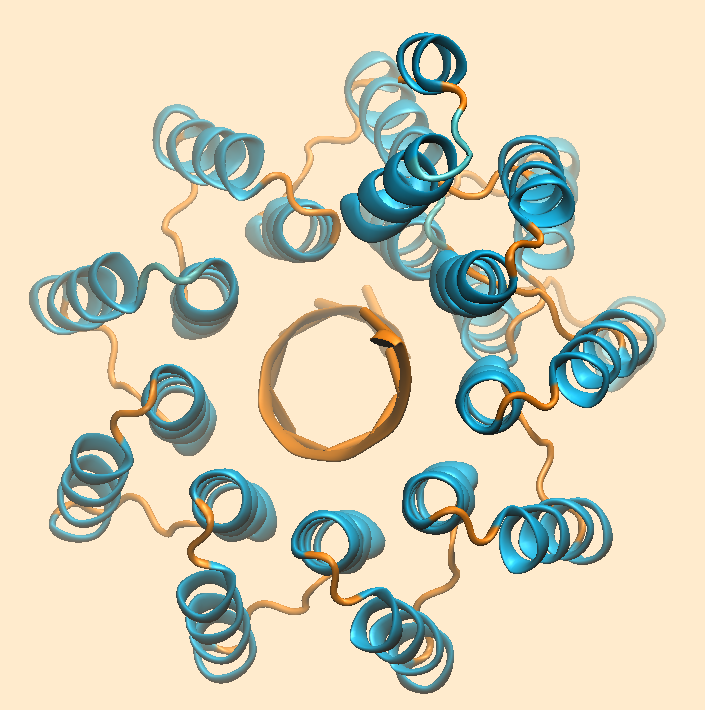
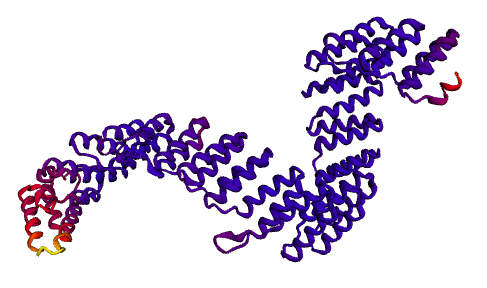
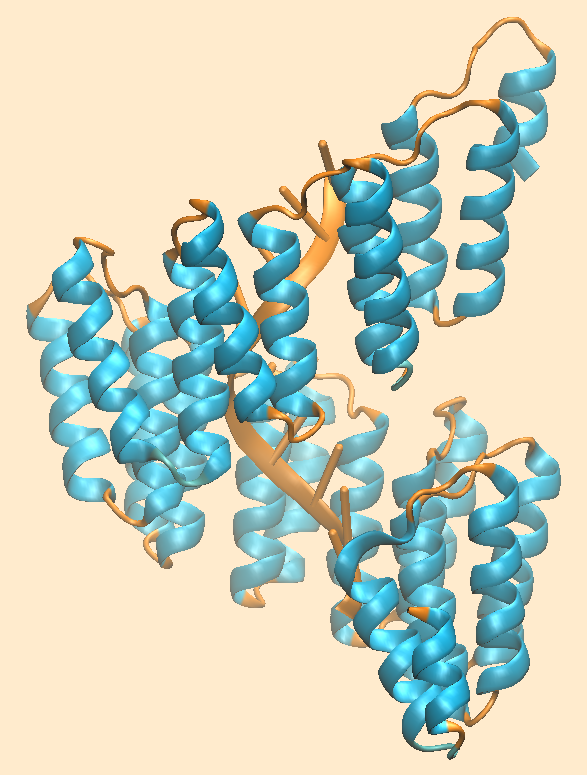
Endosymbiose 2aire

Algues rouges

**Figure 1 :** (A) Diatomées observées en microscopie à contraste de phase. (B) Arbre phylogénétique des algues marines ??

NE PAS OUBLIER DE CITER LES IMAGES, SOURCES ET ARTICLES

La régulation génomique du chloroplaste est connue chez les algues vertes, mais beaucoup moins chez les algues rouges dont font partis les diatomées. Toutefois ces organismes étant proches évolutivement parlant, nous utiliseront nos connaissances des algues vertes dans notre étude. Ainsi chez les algues vertes ce type de mécanisme s’effectue par des protéines possédant une structure particulière appelée alpha-solénoïde. Elles sont aussi connues pour interagir particulièrement avec l’ARN messager chloroplastique dans des processus de maturation, stabilisation, épissage, translation et dégradation génétique. Ces protéines en alpha solénoïdes sont adressées aux organites : mitochondrie et chloroplaste. Les motifs en alpha solénoïdes sont constitués d’une répétition d’hélices alpha en antiparallèle et conférant leur structure caractéristique aux protéines. Un motif de répétition constitue alors deux hélices antiparallèles. De part leur forme, les protéines en alpha solénoïdes se fixent sur l’ARNm chloroplastique pour le réguler. En effet, l’ARNm se positionne du côté 5’ au sein de la cavité formée par la structure en hélice alpha pour se lier à la protéine. De plus, il semblerait que la méthode de reconnaissance de l’ARNm soit due à une spécificité des bases azotées en plus d’une reconnaissance dite « one repeat :one-nucleotide ». Cela signifie que la reconnaissance nucléotidique permet à un nucléotide de se fixer entre deux motifs d’hélices.

C

A

B

**Figure 2 :** (A) Protéine en alpha solénoïde. (B) Protéine en alpha solénoïde liée à un ARNm du côté 5’, le carré noir montrant 1 motif de répétition d’hélice alpha en antiparallèle. (C) De même que la figure 2B, la flèche représentant le positionnement des nucléotides entre deux motifs : c’est ce que l’on appelle la reconnaissance par « one repeat : one nucleotide ».

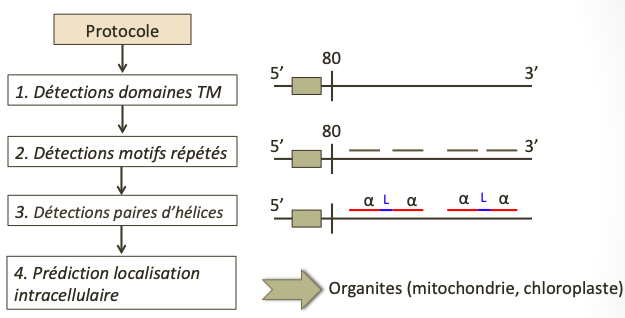
1. Matériel et méthodes

Les codes permettant de répondre aux questions et problématiques ont été rédigés essentiellement en python (version 3.9), puis en R.

Nous cherchons à pouvoir identifier des protéines en alpha solénoïdes impliquées dans la régulation post-transcriptionelle sur la base de leurs propriétés structurales et physico-chimiques. Nous savons que ces protéines ont des caractéristiques communes :

* Elles sont adressées au choloroplaste ou à la mitochondrie.
* Elles ne possèdent pas de domaines transmembranaires après le 68e acides aminé (s’il y en a avant cela correspondrait à un peptide d’adressage, ce qui est en accord avec nos attentes).
* De part leur structure en hélice α elles possèdent des sites ou acides aminés dit « linker » situés entre deux hélices et faisant le lien; mais aussi des répétitions de séquences caractéristiques de ces hélices.

Pour déterminer les possibles candidats à la régulation post-transcriptionelle nous disposons d’un protocole au préalablement conçu qui permet d’identifier ces protéines à l’aide de différentes étapes et d’outils de prédictions utilisés en stand alone.



68

68

68

Domaine transmembranaire

Figure x : Protocole mis en place pour identifier les protéines en alpha solénoïdes

Ces outils et leurs prédictions sont visibles dans le tableau ci-dessous.

À la suite de cela nous pouvons identifier des protéines candidates à la régulation post-transcriptionelle en alpha solénoïdes. Il faut noter que l’on utilise plusieurs outils de prédiction d’adressage car il peut y avoir des biais de résultats selon les outils. En combinant les soties obtenues, nous aurons donc des résultats plus fiables en diminuant le biais. Chaque logiciel prends en entrée un protéome, et a un output (ou sortie) formaté qu’il sera ensuite possible de parser (parcourir à l’aide d’un script) pour récupérer les informations qui nous intéressent.

|  |  |  |
| --- | --- | --- |
| **Outil** | **Prédiction** | **Information** |
| TMHMM | Structure transmembranaire | Identifiants des protéines sans domaine transmembranaire (après le 68e acide aminé) |
| Targetp2 | Présence de pré-séquence en Nter (peptide d’adressage) | Adressage[[1]](#footnote-1) |
| Ard2 | Acide aminés linker d’hélice α | Nombre de linker, probabilités d’être linker |
| Radar | Répétitions de séquences | Nombre de répétitions, longueur et fréquence d’acide aminé pour chaque répétition, proportion de la séquence qui possède des répétitions |
| Deeploc | Adressage | Probabilité d’adressage pour la mitochondrie et le chloroplaste |
| Wolfpsort | Adressage | Identifiants des protéines adressées à la mitochondrie ou au chloroplaste |
| Localizer | Adressage | Identifiants des protéines adressées au chloroplaste |

Table x : nfiezongeizgneizneiz

+parler de ce qu’on obtient en output

+ le stand alone 🡪 en quoi ça consiste et comment je les ai utilisés

Nous utilisons donc 7 logiciels :

Le logiciel TMHMM permet de prédire la présence et la position de segments transmembranaires. Il fonctionne sur le modèle de chaînes cachées de Markov-Nikov. Il permet ainsi de déterminer si une protéine est transmembranaire ou non, nos protéines d’intérêt ne l’étant pas. L’output de TMHMM est un fichier qui nous donne pour chaque protéine du protéome le nombre de segments transmembranaires prédits ainsi que leur positions.

Targetp2 quant à lui est un modèle prédictif de deep learning, entraîné et testé sur des modèles de séquences de 200 acides aminés construits à partir de la matrice de substitution BLOSUM62. Il sert à détecter des séquences dites « signales » dans l’adressage protéique. Ces séquences sont propres à un adressage qui peut être dans le cas qui nous intéresse mitochondrial ou chloroplastique. En sortie, Targetp2 écrits un fichier pour le protéome d’entrée qui contient pour chaque protéine l’adressage qui a été déterminé.

Le logiciel Ard2 (Alpha-rod Reapeat Detector) rend possible l’identification de sites de linker entre hélices alpha, permettant ainsi la détection de protéines possédant des structures en alpha solénoïdes en utilisant un réseau neuronal. Ce logiciel nous donne pour chaque acide aminé de chacune des séquences la probabilité associée d’être un linker.

RADAR (Rapid Automatic Detection adn Alignment of Repeats) permet la détection de motifs répétés complexes au sein de séquences protéiques. Ici, cela nous permettra de potentiellement repérer la présence d’hélices α qui sont des structures aux motifs répétés. Ce logiciel fonctionne sur un algorithme capable d’identifier par approximation des répétitions de motifs et de structures de différents types. En sortie radar écrit un fichier contenant principalement pour chaque séquences du protéome : le nombre de répétitions trouvées, leurs positions et leur séquence.

Quant à Deeploc, Wolfpsort et Localizer ils permettent la prédiction d’adressage d’une protéine. En effet Deeploc distingue 10 localisations cellulaires différentes[[2]](#footnote-2) et sur la base de la séquence permet d’associer une probabilité d’adressage pour chaque localisation à une protéine. Il fonctionne grâce à des réseaux neuronaux récurrents.

Wolpsort utilise la composition en acide aminé qu’il convertit en vecteurs pour être ensuite classifiés par méthode k-nearest neighbor, qui est un algorithme supervisé de machine learning qui peut être utilisé à des fins de classification ou de régression. L’algorithme permet donc de déterminer la localisation cellulaire d’une protéine par l’intermédiaire de sa proportion en acide aminé. En sortie wolfpsort nous rends un fichier contenant un identifiant protéique par ligne dont l’adressage est noté par un score puis ordonné de manière décroissante, le total étant noté sur 14.

Enfin, l’algorithme de Localizer est une méthode de machine learning qui a été entraîné pour prédire la localisation des protéines chez les plantes. L’adressage vers le chloroplaste ou la mitochondrie est prédite en déterminant la présence de peptide signal (ou peptide de transit) tandis que la localisation vers le noyau est prédite par un ensemble de signaux peptidiques nucléaires (NLSs). Pour ce qui est du fichier de sortie, on obtient essentiellement par ligne l’identifiant protéique, et la première colonne correspondant à l’adressage au chloroplaste. Si la protéine est identifiée comme étant ciblée pour cet organite alors la première colonne sera marquée d’un ‘Y’.

Tous ces logiciels ont été utilisés en stand alone et non en libre service sur internet. C’est-à-dire que nous avons utilisé directement les codes et script sources -🡪 mettre lignes de commandes écrites dans le terminal

En premier lieu deux protéomes ont été fournis. Ces protéomes contiennent les séquences au format fasta de protéines que l’on sait être en alpha solénoïde (pour le protéome de témoin positif) et de protéines qui ne le sont pas (pour le protéome de témoin négatif). Ces protéomes ont été obtenus à l’issu du travail d’un ancien stagiaire de M2.

*Lancement des logiciels et parsing des fichiers :*

🡪 Comment j’ai fais tourner les lignes de commandes et les fichiers

Nous avons donc fait tourner d’abord TMHMM sur ces protéomes et récupéré en effectuant un parsing des output les identifiants des protéines ne possédant pas de domaines transmembranaires après le 68e acide aminé. Ensuite nous avons reconstitué de nouveaux protéomes positifs et négatifs avec les séquences des protéines sélectionnées à l’issu du filtrage. Ce sont ces nouveaux protéomes qui ont ensuite été donnés comme input aux autres logiciels. Il est donc important de commencer par utiliser TMHMM, même l’ordre d’utilisation des autres logiciels reste négligeable. Chacun des autres logiciels a donné des outputs que l’on a filtré pour ne garder que l’essentiel, visible ci-dessus dans le tableau récapitulatif (table x).

Pour ce qui est du parsing de targetp2, nous avons récupéré les résultats d’adressage prédit pour chaque protéine, que nous avons stocké dans une dictionnaire.

En première sortie ard2 réécrit le protéome pour le formater. Ensuite il faut redonner ce premier output en entrée à ard2 pour que l’output final soit accessible. Celui-ci donne pour chaque séquence la probabilité de chaque acide aminé d’être un linker. Ainsi nous sélectionnons les acides aminés donc la probabilité est supérieure à 0.10. De plus, si l’on dispose d’une « région de linker » nous considérons qu’il faut sélectionner l’acide aminé avec la plus grande probabilité de linker. Pour cela nous avons utilisé le principe de fenêtre glissante. Cette fenêtre parcourt chaque acide aminé et son résultat et est constituée de 6 acides aminés. Au sein de cette fenêtre nous prenons l’acide aminé qui possède la plus forte probabilité lorsqu’elle est supérieure à 0.10, ainsi que sa position au sein de la séquence.

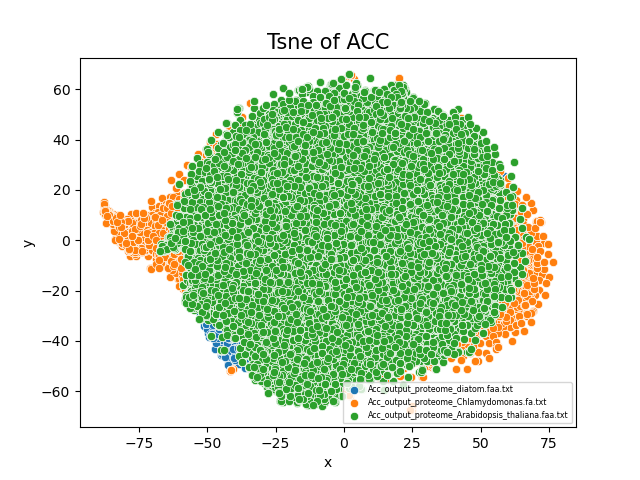
Rappeler but de l’étude

* les outils de prédictions v
* protéomes à disposition v
* témoins positifs/ négatifs 🡪 ce que c’est, d’où ca vient, ce que j’ai fait avec, le parsing ect
* la même chose avec les protéomes de diatomées que céline m’a fournit
* Acc et freq

Mettre en annexe les résultats output des logiciels + la dataframe finale

1. Résultats



**

🡪 on veut discriminer les protéomes 🡪 est-ce qu’ils y a des différences significatives ? 🡪 si non on peut continuer l’analyse dessus

ici l’acc semble ne pas être différent sur chacun des protéomes

***RÉFÉRENCES***

[] Guo & al. (2022) « A computational method for predictiong nucleocapsid protein in retroviruses », *Scientific Reports, Volume 12, page 524, 10.1038/s41598-021-03182-2.*

[] Sandaruwan & al. (2021) « An improved deep learning model for hierarchical classification of protein families », *PloS ONE, Volume 16, Numéro 10, 10.1371/journal.pone.0258625*.

[] Zhao & al. (2021) « DescribePROT : database of amino acid-level protein structure and function predictions », *Nucleic Ancids Reasearch, Volume 49, Numéro D1, pages D298-D308, 10.1093/nar/gkaa931.*

[] Chatzimparm & al. (2020) « t-viSNE : Interactive Assessment and Interpretation of t-SNE Projections », *IEE Trans. Visual. Comput. Graphics, Volume 26, Numéro 8, pages 2696-2714, 10.1109/TVCG.2020.2986996.*

1. Sorties possibles : SP = signal peptide, mTP = mitochondrial transit peptide, cTP = chloroplast transit peptide, iTP = thylakoid transit peptide, NoTP = aucun signal [↑](#footnote-ref-1)
2. Noyau, Cytoplasme, extracellulaire, mitochondriale, membranaire, reticulum endoplasmique, chloroplaste, appareil de Golgi, Lysosome/vacuole et perixysome. [↑](#footnote-ref-2)