Faculté d'informatique – Département IA &SD

"Rapport sur Alignement Multiple de Séquences "

Bio-informatique : Bio-Algorithmique

Par : Bouhraoua Yousra Hind et
Hamdi Pacha Aya

Année : 2023/2024

Sommaire

Introduction

La bio-informatique est un domaine multidisciplinaire qui fusionne la biologie, l'informatique, les mathématiques et les statistiques. Son objectif principal est d'analyser les séquences biologiques et de prédire la structure et la fonction des macromolécules, telles que les protéines et les acides nucléiques. Cette discipline s'est développée pour répondre aux besoins croissants dans divers domaines tels que l'agriculture, la pharmacologie et la médecine. Elle évolue constamment pour relever de nouveaux défis posés par la biologie.

Un aspect crucial de la bio-informatique est la manipulation de grandes quantités d'informations génétiques, telles que les séquences d'ADN et d'ARN, l'alignement de séquences est une technique cruciale pour comparer et analyser ces séquences biologiques,il permet de mettre en évidence les similitudes et les différences entre ces séquences, ce qui est essentiel pour comprendre leur évolution, leur fonction et leur structure.

Importance de l'Alignement

- **Comprendre l'Évolution:** L'alignement de séquences est fondamental pour étudier l'évolution des organismes. Il permet de reconstruire l'historique évolutif en identifiant les séquences homologues (c'est-à-dire ayant une origine commune) et en analysant leurs mutations.

- **Identification de Sites Fonctionnels:** Les régions conservées dans les séquences alignées sont souvent associées à des sites fonctionnels importants, tels que les sites catalytiques ou les zones d'interaction. Ces régions sont sous une forte pression de sélection, ce qui signifie qu'elles sont conservées tout au long de l'évolution.

- **Prédiction de Structures et de Fonctions:** L'alignement de séquences peut aider à prédire la structure tridimensionnelle d'une protéine si celle-ci est similaire à une protéine dont la structure est déjà connue. De plus, il peut également prédire la fonction d'une protéine en identifiant des motifs ou des domaines fonctionnels présents dans des séquences connexes.

- **Phylogénie:** L'alignement multiple permet de construire des arbres phylogénétiques, qui illustrent les relations évolutives entre différents organismes ou groupes d'organismes. Cela aide à comprendre comment les différentes espèces se sont évoluées et se sont divergées au fil du temps.

Types d'Alignement

Alignement Par Paires: Compare deux séquences à la fois, soit globalement (sur toute leur longueur), soit localement (entre une séquence et une partie de l'autre). Des algorithmes comme Needleman-Wunsch pour

l'alignement global et Smith-Waterman pour l'alignement local sont couramment utilisés.

Alignement Multiple: Aligne trois ou plusieurs séquences simultanément pour identifier les régions conservées et les divergences. Cela est particulièrement utile pour étudier des familles de protéines ou des groupes d'ADN/ARN.

Dans ce projet on va implémenter et analyser les 2 types d'alignement de séquences

Notions de Base

Les séquences biologiques, telles que celles d'ADN et d'ARN, jouent un rôle central dans la transmission et l'expression de l'information génétique dans tous les êtres vivants.

Séquences d'ADN

- **Structure de l'ADN:** L'ADN est une double hélice formée de deux brins enroulés autour d'un axe central. Chaque brin est composé de nucléotides, qui sont des unités de construction de l'ADN. Les nucléotides contiennent une base nitrogenée (adénine, cytosine, guanine, ou thymine) et un sucre (deoxyribose) .

- **Bases Complémentaires:** Les bases d'ADN se lient selon des paires complémentaires : adénine (A) avec thymine (T), et cytosine (C) avec guanine (G). Cette règle assure la stabilité de la double hélice.

- **Rôle de l'ADN:** L'ADN stocke l'information génétique nécessaire à la reproduction et au développement des organismes. Sa séquence codée contrôle la synthèse de protéines, qui sont les effecteurs de la plupart des fonctions cellulaires.

Séquences d'ARN

- **Types d'ARN:** Il existe plusieurs types d'ARN, dont l'ARN messager (mRNA), l'ARN ribosomal (rRNA), et l'ARN transférant (tRNA). Chaque type d'ARN a une fonction spécifique dans le processus de synthèse des protéines.

- **Composition:** Tous les ARNs sont constitués de nucléotides, similaires à ceux de l'ADN, mais avec une petite différence : l'ARN contient uracile (U) au lieu de thymine (T) présent dans l'ADN.

- **Transcription:** La transcription est le processus par lequel l'information de l'ADN est transmise à l'ARN. L'ARN messager est produit à partir d'un segment spécifique de l'ADN appelé gène. Ce processus est catalysé par l'ARN polymérase.

- **Traduction:** Après la transcription, l'ARN messager quitte le noyau et migre vers les ribosomes dans le cytoplasme. Ici, il sert de template

pour la synthèse de protéines, où chaque triplet de bases de l'ARN messager est traduit en un acide aminé spécifique.

En résumé, les séquences d'ADN et d'ARN sont essentielles pour la transmission et l'expression de l'information génétique. Elles jouent un rôle central dans la synthèse des protéines, qui sont les effecteurs de la plupart des fonctions vitales.

l'Alignement de Séquences

L'alignement de séquences consiste à superposer deux ou plusieurs séquences biologiques de manière à maximiser le nombre de correspondances entre les éléments (nucléotides pour l'ADN et l'ARN, acides aminés pour les protéines). Cela permet d'identifier les régions homologues ou similaires, qui sont souvent associées à des fonctions biologiques communes. L'alignement introduit des "trous" pour compenser les différences de taille entre les séquences, reflétant des insertions ou des suppressions (appelées indels)

Exemple d'alignement simple:

Considérons deux séquences d'ADN simples:

```
Séquence A: AGCTAGCTAGCT
Séquence B: AGCTAGCTAGCTG
```

Pour aligner ces deux séquences, nous pouvons introduire un trou(-) après le nucléotide de la Séquence B pour maximiser les correspondances:

```
AGCTAGCTAGCT
AGCTAGCTAGCT-
```

En pratique lors de l'utilisation des algorithmes d'alignement , on utilise des **Matrices de similarités** qui vont nous fournir des scores de similarité entre les paires d'éléments (nucléotides pour l'ADN et l'ARN, acides aminés pour les protéines), facilitant ainsi la comparaison directe des séquences.

- **Matrices de Dayhoff (PAM):** Basées sur des distances évolutives entre espèces, elles sont conçues pour représenter les probabilités acceptables de mutation entre les acides aminés. Ces matrices sont nommées PAM (Probability of Acceptable Mutations) et varient en termes de tolérance aux substitutions d'acides aminés, offrant ainsi différentes options pour l'alignement de séquences.

- **Matrices de Henikoff (BLOSUM):** Contrairement aux matrices PAM, les matrices BLOSUM sont basées sur le contenu en information des substitutions. Elles ont été développées pour être plus robustes face aux variations dans les séquences biologiques, en particulier dans les protéines. Les matrices BLOSUM existent en plusieurs versions, chacune adaptée à des niveaux différents de divergence entre les séquences, allant de BLOSUM62 (pour des séquences très proches, qu'on va utiliser dans ce projet) à BLOSUM90 (pour des séquences très distantes).

Alignement par Paire

L'alignement par paire vise à comparer deux séquences biologiques, telles que l'ADN, l'ARN ou les protéines, afin de déterminer leur degré de similarité. Cependant ces séquences nécessitent très souvent l'alignement de séquences longues, très variables ou extrêmement nombreuses, qui ne peuvent pas être alignées manuellement ou par une exploration exhaustive de tous les alignements possibles. En effet, le nombre d'alignements possibles pour une paire de séquences augmente factoriellement avec la taille des séquences. Pour deux séquences de taille respective n et m, il existe $\frac{(m+n)!}{m!\times n!}$ alignements possibles. Pour faciliter le processus d'alignement on peut trouver des algorithmes exacts.

Alignement Local

Se concentre sur les régions de meilleure similarité entre les deux séquences, sans exiger une correspondance continue sur toute la longueur des séquences. Cela est particulièrement utile pour identifier des motifs ou des domaines fonctionnels spécifiques dans des protéines multi-domaines. L'algorithme de **Smith-Waterman** est souvent utilisé pour cet alignement local, en recherchant des segments de séquences qui sont significativement similaires.

Alignement Global

Assure que chaque partie des deux séquences est alignée, ce qui est réalisé grâce à l'algorithme de Needleman-Wunsch.

Needleman-Wunsch

L'algorithme Needleman-Wunsch a été publié en 1970. C'est un algorithme de programmation dynamique qui utilise une matrice de similarité pour attribuer des scores positifs aux correspondances, négatifs aux substitutions et pénalités pour les trous (gaps) introduits pour aligner les séquences. L'objectif est de trouver l'alignement qui maximise le score total, reflétant la similarité maximale entre les deux séquences.

Principe de l'Algorithme:

1. **Initialisation :**
    - Créer une matrice de score de dimensions (m+1) x (n+1), où m et n sont les longueurs des deux séquences à aligner.

    - Initialiser la première ligne et la première colonne de la matrice avec des multiples de la pénalité d'indel, représentant l'alignement de la séquence avec des gaps.
2. **Remplissage de la Matrice :**

    Pour chaque cellule S(i,j) de la matrice on prend la valeur maximale entre:
    - la cellule d'en haut S(i-1,j)+ indel

- la cellule d'a gauche S(i,j-1)+ indel
- la cellule S(i-1,j-1) + cout()

3. **Traceback :**
    - Partir de la dernière cellule (m, n) de la matrice et retracer le chemin optimal jusqu'à la première cellule (0, 0).
    - Les directions possibles (diagonale, haut, gauche) déterminent si l'alignement est une insertion,une suppression ou une substitution.
        - **Diagonale (substitution) :** Si $S(i,j)$provient de S(i−1,j−1)+cout(seq1[i−1],seq2[j−1]), alors inclure les caractères correspondants des deux séquences dans l'alignement.
        - **Vers le haut (insertion) :** Si S(i,j) provient de S(i−1,j)+indel, alors inclure un gap ('-') dans la séquence 2.
        - **Vers la gauche (insertion) :** Si S(i,j) provient de S(i,j−1)+indel, alors inclure un gap ('-') dans la séquence 1.

Exemple d'Alignement:

Séquence 1 : NEEDLEMAN

Séquence 2 : NEALDLMAN

Nous utiliserons la matrice BLOSUM62 et une pénalité de -4.

		N	E	E	D	L	E	M	A	N
	0	-4	-8	-12	-16	-20	-24	-28	-32	-36
N	-4	6	2	-2	-6	-10	-14	-18	-22	-26
E	-8	2	11	7	3	-1	-5	-9	-13	-17
A	-12	-2	7	10	6	5	1	-3	-7	-11
L	-16	-6	3	6	6	12	8	4	0	-4
D	-20	-10	-1	2	10	8	16	12	8	4
L	-24	-14	-5	-2	6	12	12	14	11	8
M	-28	-18	-9	-6	2	8	14	17	13	9
A	-32	-22	-13	-5	-2	4	10	13	21	17
N	-36	-26	-17	-9	-6	0	6	9	17	27

L'alignement optimal obtenu est :

N E E - D L E M A N

N E A L D L - M A N

Cas d'Utilisation de l'Alignement par Paires

L'alignement par paires de séquences est un outil fondamental en bio-informatique, avec de nombreuses applications pratiques et scientifiques. Voici quelques cas d'utilisation principaux :

- **Études Évolutives :** Comprendre les relations phylogénétiques entre différentes espèces ou entre différents gènes au sein d'une même espèce.
- **Annoter des Génomes :** Identifier les gènes dans un nouveau génome en le comparant à un génome de référence.
- **Recherche de Gènes/Protéines dans des Bases de Données de Séquences :** BLAST (Basic Local Alignment Search Tool) est un exemple bien connu qui utilise l'alignement par paires pour trouver des séquences similaires dans de grandes bases de données.
- **Découverte de Nouveaux Gènes :** Identifier des gènes inconnus en les alignant avec des gènes connus.
- **Fonction des Protéines :** Prédire la fonction d'une nouvelle protéine en la comparant avec des protéines bien caractérisées.
- **Génomique Médicale :** Identifier les mutations génétiques associées à des maladies.

Implémentation

1-Initialisation :

```
def needleman_wunsch(seq1,seq2,indel,matrix):
  n,m=len(seq1),len(seq2)
# Initialisation de la matrice des scores
  score_matrix = np.zeros((n+1, m+1))
  for j in range(m+1):
      score_matrix[j][0] = j * indel
  for i in range(n+1):
      score_matrix[0][i] = i * indel
```

2-Remplissage de la Matrice:

```
# Remplissage de la matrice des scores
  for i in range(1,n+1):
    for j in range(1,m+1):

diag=score_matrix[i-1][j-1]+int(matrix[seq1[i-1]][seq2[j-1]])
      gauche=score_matrix[i][j-1]+indel
      haut=score_matrix[i-1][j]+indel
      score_matrix[i][j]=max(diag,gauche,haut)
```

3-Traceback(Alignement):

```
  # Traceback : retrouver l'alignement optimal
  seq_al1=''
  seq_al2=''
  i,j=n,m

  while i>0 and j>0:
    score=score_matrix[i][j]
    # si la case provient de la diagonale
```

```
    if score==score_matrix[i-1][j-1] +
int(matrix[seq1[i-1]][seq2[j-1]]):
      seq_al1+=seq1[i-1]
      seq_al2+=seq2[j-1]
      i-=1
      j-=1
    #si elle provient d'en haut
    elif score==score_matrix[i-1][j]+indel:
      seq_al1+=seq1[i-1]
      seq_al2+='-'
      i-=1
    # si elle provient d'a gauche
    else:
      seq_al2+=seq2[j-1]
      seq_al1+='-'
      j-=1
  while i > 0:
        seq_al1 += seq1[i-1]
        seq_al2 += '-'
        i -= 1
  while j > 0:
        seq_al1 += '-'
        seq_al2 += seq2[j-1]
        j -= 1
```

4- Exemple:

```
seq1 = "NEALDLMAN"
seq2 = "NEEDLEMAN"

alignment = needleman_wunsch(seq1, seq2,-4,blosum)
print(f"Alignement:\n{alignment[0]}\n{alignment[1]}\nScore:
{alignment[2]}")
```

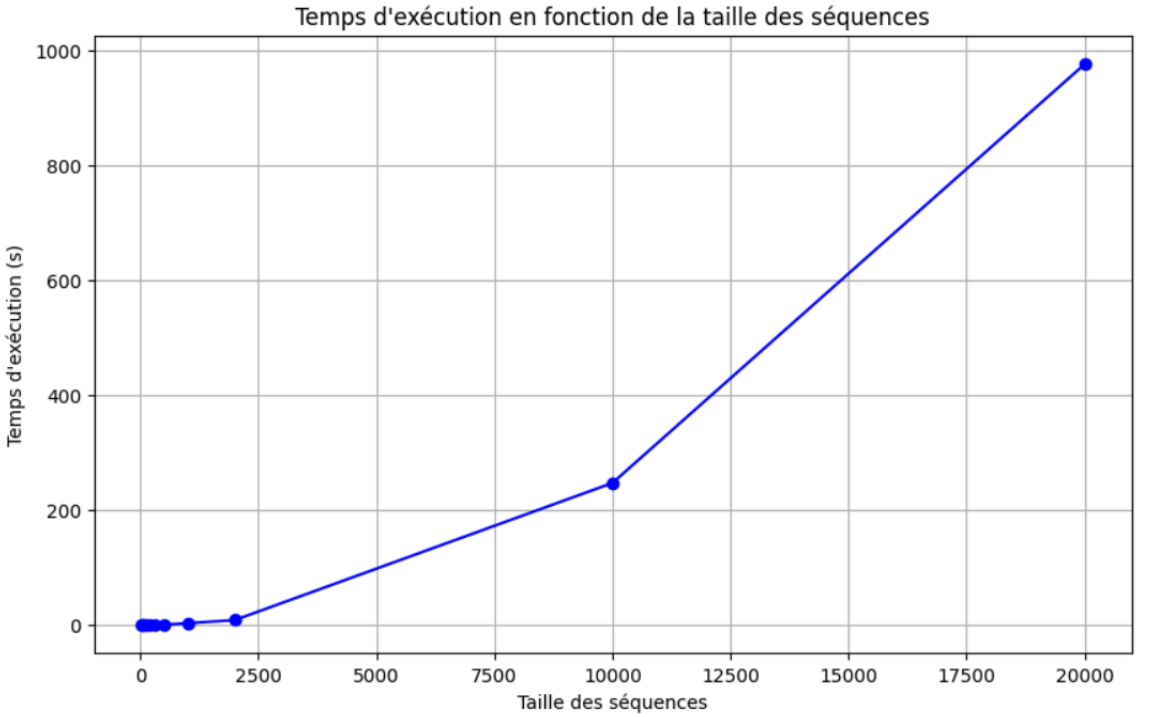
```
Alignement:
NEALDL-MAN
NEE-DLEMAN
Score: 27
```

Tests et Expérimentations

On a généré des séquences de longueur 25 ,50 ,100 ,200 ,300 ,500… pour tester Needleman-wunsch et voila les résultats obtenus:

	Length	Score	Execution Time (s)
0	25	53	0.002244
1	50	112	0.009681
2	100	276	0.019840
3	200	622	0.078775
4	300	897	0.195037
5	500	1554	0.523571
6	1000	3180	3.426914

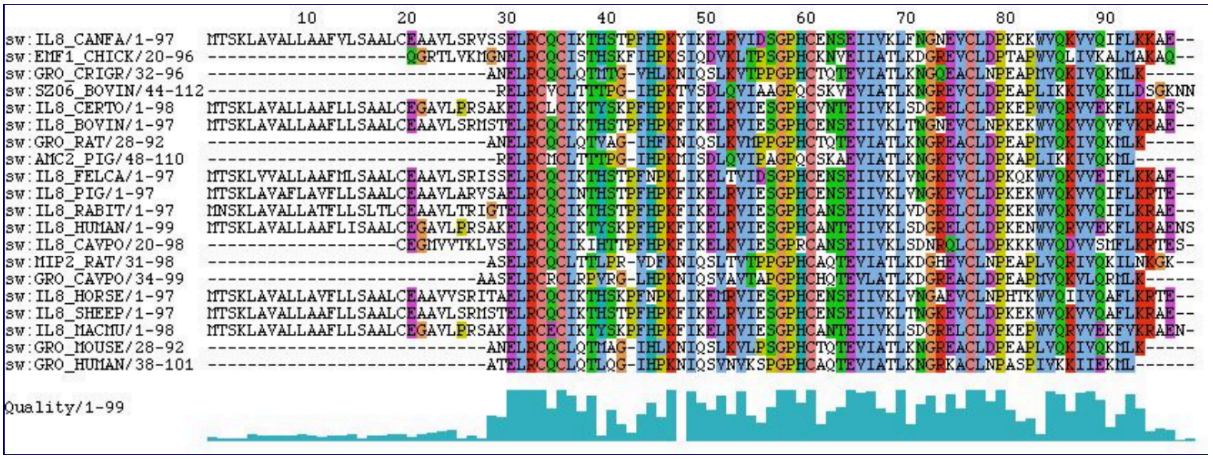
On remarque que le temps d'exécution à augmenter dès qu'on est passé de 2000 à 10000 (voir le graphe).



Alignement Multiple

Les méthodes d'alignement présentées ci-dessus permettent d'identifier ce qui est commun à deux séquences. Lorsque l'on veut identifier ce qui est conservé dans trois séquences ou plus, ces méthodes ne suffisent plus. C'est pourquoi de nombreuses recherches ont été menées pour développer une méthode automatique d'alignement multiple sur la seule base des séquences tout en essayant de tenir compte des contraintes physico-chimiques des éléments et les structures des séquences. Nous pouvons distinguer deux groupes de méthodes, les approches itératives et les approches progressives. Les approches progressives sont probablement encore les méthodes les plus utilisées actuellement.

Exemple d'alignement multiple:



Cas d'Utilisation de l'Alignement Multiple

L'alignement multiple de séquences est une technique puissante en bioinformatique qui trouve de nombreuses applications dans divers domaines scientifiques, notamment en génomique, en protéomique et en évolution moléculaire. Voici quelques cas d'applications clés :

- **Identification de Régions Conservées :** L'alignement multiple permet d'identifier les régions très conservées entre les séquences, qui sont souvent associées à des fonctions biologiques importantes.

- **Comparaison de Protéines et d'ARN :** L'alignement multiple permet de comparer des protéines et des ARN entre différentes espèces, facilitant ainsi l'identification des variations et des adaptations évolutives. Cela est particulièrement utile pour étudier les mécanismes de développement, la pathogenèse des maladies et l'évolution des traits adaptatifs.

- **Analyse Structurale et Fonctionnelle :** Les alignements multiples peuvent aider à prédire la structure tridimensionnelle des protéines et à identifier des sites fonctionnels spécifiques, tels que les sites de liaison ou les sites catalytiques.

Approche Exacte (Programmation Dynamique)

L'approche exacte utilise la programmation dynamique pour trouver l'alignement optimal de plusieurs séquences. L'algorithme de Needleman-Wunsch, par exemple, peut être étendu pour aligner plus de deux séquences. La programmation dynamique construit une matrice n-dimensionnelle (où n est le nombre de séquences), en remplissant cette matrice pour trouver l'alignement global optimal.

- **Avantages :**
    - Optimalité : Garantit de trouver l'alignement global optimal.
    - Théorique : Utile pour des comparaisons théoriques et des validations d'autres méthodes.
- **Limites :**
    - Complexité Computationnelle : La complexité en temps et en espace est exponentielle (O(L^n) pour n séquences de longueur L), ce qui le rend impraticable pour plus de trois séquences de longueur modérée.
    - Mémoire : Exige une énorme quantité de mémoire pour stocker la matrice n-dimensionnelle.

Approche Progressive (Notion de Profil)

L'approche progressive (utilisée dans ClustalW) commence par aligner les paires de séquences les plus similaires et ajoute progressivement les autres séquences à l'alignement. Cela implique la construction d'un arbre guide (guide tree) basé sur les distances ou les similarités entre les séquences. Ensuite, les séquences ou groupes de séquences sont alignés selon l'ordre déterminé par l'arbre guide. Chaque groupe de séquences alignées est représenté par un profil.

- **Avantages :**
    - Efficacité : Plus rapide que l'approche exacte et pratique pour un grand nombre de séquences.
    - Flexibilité : Peut gérer un grand nombre de séquences de différentes longueurs.
- **Limites :**
    - Sous-optimalité : Le résultat final peut ne pas être l'alignement global optimal, car des erreurs initiales d'alignement peuvent être propagées.
    - Dépendance à l'arbre guide : La qualité de l'alignement dépend de la précision de l'arbre guide initial.

Approche Itérative (DiAlign, SAGA)

L'approche itérative commence par un alignement initial, puis améliore cet alignement par des cycles successifs de ré-alignement. Des méthodes comme DiAlign et SAGA utilisent des heuristiques pour réviser et ajuster continuellement l'alignement afin d'optimiser un score global. L'alignement est raffiné à chaque itération pour corriger les erreurs et améliorer la qualité globale.

- **Avantages :**
    - Amélioration Continue : Peut améliorer un alignement initial sous-optimal à travers des itérations successives.
    - Adaptabilité : Peut être utilisé pour affiner les résultats obtenus par d'autres méthodes comme l'approche progressive.
- **Limites :**
    - Temps de Calcul : Peut être computationnellement coûteux et prendre beaucoup de temps, surtout pour de longues séquences ou un grand nombre de séquences.
    - Dépendance à l'initialisation : La qualité du résultat final peut dépendre de l'alignement initial

Chacune de ces approches a ses avantages et ses inconvénients, et le choix de la méthode dépend des besoins spécifiques de l'analyse, notamment en termes de précision requise, de taille des données et de ressources disponibles.

Implementation

- On a testé avec plusieurs nombres de séquences de taille 50 voila quelque alignement obtenue

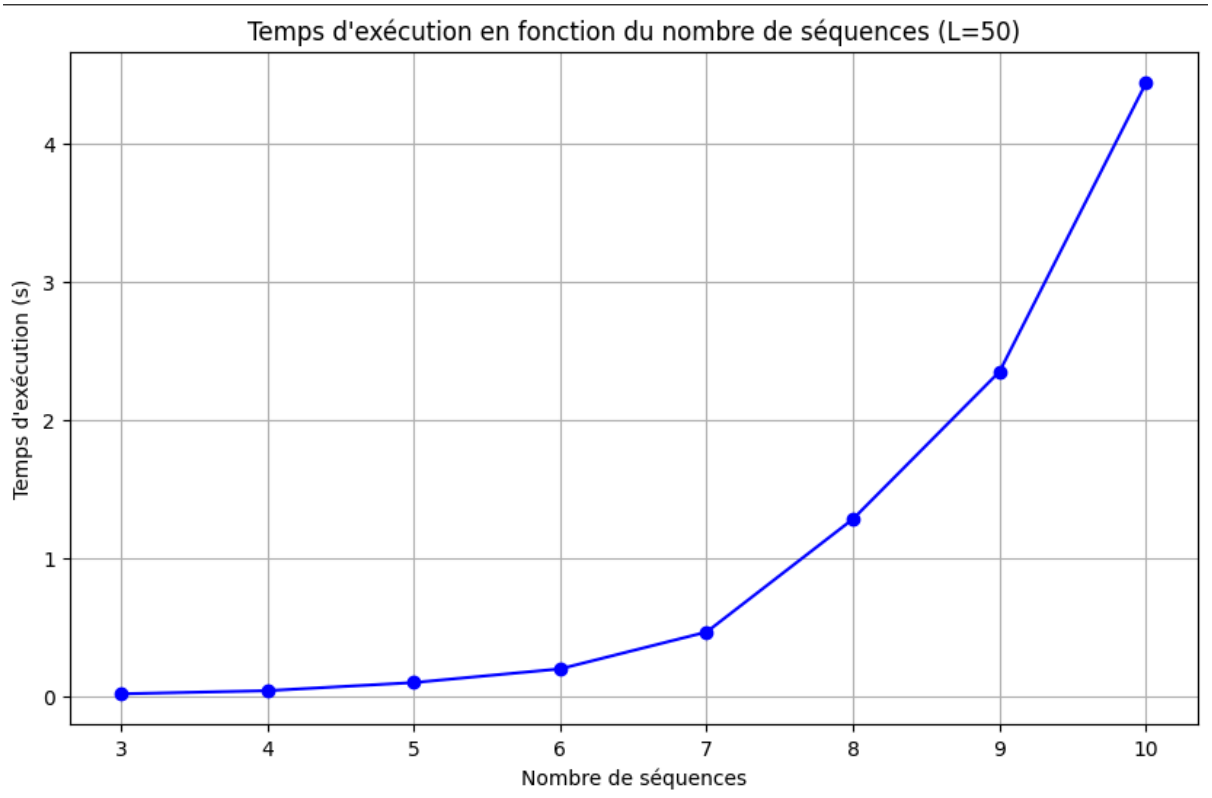
```
Matrice de distance:
 [[ 0. 41. 37.]
 [41.  0. 41.]
 [37. 41.  0.]]

Alignement pour 3 séquences:
séquence 1: --TTGTTGCTTCTAA---TCAAGTATTG-CAC-TGT-GC-GATAGCGTC--CACCGGC-TGC
séquence 2: GGTGGTTG-GGGTAACCTTCCA--ATTG-GACAT-CCCCCG--GGC-T-TTCA--GAC-AAC
séquence 3: ACCTGGCCCCT-GGG-AA--ATTGTC-C-GT-T--ACCTGAT--C-TTCTTTAATATCATAT
séquence 2: -GGTGG---TTGGGGTAA--CCT-TC-CAAT-TGGACAT-CCCCCGGGCTTTCAGA-CA-AC
Matrice de distance:
 [[ 0. 34. 38. 39.]
 [34.  0. 38. 41.]
 [38. 38.  0. 30.]
 [39. 41. 30.  0.]]

Alignement pour 4 séquences:
séquence 3: -TAGT---A-A-GT-CCCG-A--A-GGC-GATGTCCCTG-ATTATGGCCCGA--TT-TG-CGAGTGG
séquence 2: --A-T--CTCGCGT-CTAG-ACTA-GTCAGACGT---TGCA-TACGG---GTGGTTAT-TC-CCCGG
séquence 1: GT-TTC-CCACAAGT-ACT-GTTCTTG-CGAT-T-C-T-C--GT--TTAT-AACC----TGGGACCA-T--
séquence 2: AT-CTCGCGTCTAG--ACTAG-TC-AGACG-T-TGCATACGGGTGGTTAT--TCC---------CC---GG
séquence 4: CAG-AACGTGT-CTA-AACT-CCG-TG-C-AGA-GATTTTAATTC-GAT-GTT-TGCTA--AA
séquence 2: -A-TCTCGCGT-CTA-GACT--AG-T--C-AGACG--TTGCATACGGGTGGTTATTC-CCCGG
séquence 1: -------GTTTC-CCACAAGTACT-G-T---TC-TTGCGATTCTCGT--T-TATAACCTGGGACCAT
séquence 2: -------ATCTCGCGTCTAG-ACTAG-TCAGACGTTGC-ATACGGGTGGT-TAT-TCC-----CCGG
Matrice de distance:
 [[ 0. 40. 43. 39. 34.]
 [40.  0. 39. 39. 43.]
 [43. 39.  0. 38. 45.]
 [39. 39. 38.  0. 36.]
 [34. 43. 45. 36.  0.]]
```

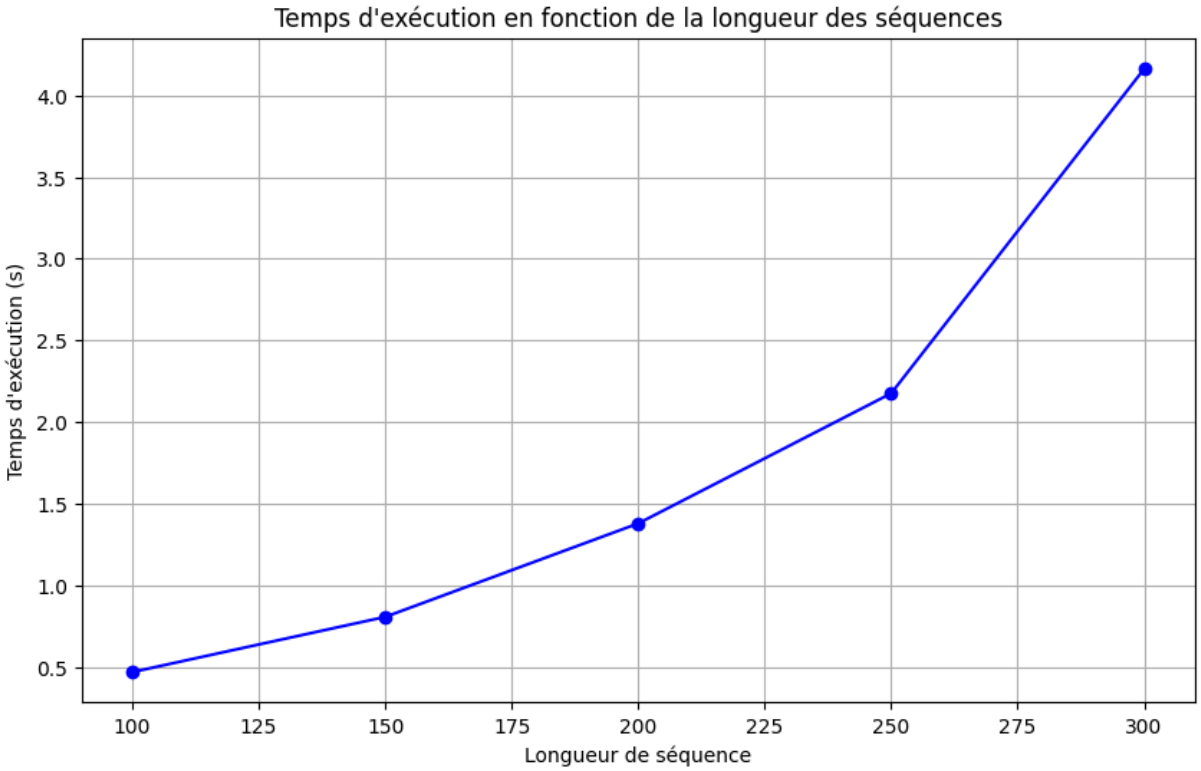
on a utilisé la distance de hamming pour créer la matrice des distances et avoir l'ordre dans lequel les séquences doivent être alignées.

Afin d'observer le changement dans le temps d'exécution , on a tracé un graphe de la variation du temps d'exécution par rapport au nombre de séquence



le temps d'exécution augmente avec l'augmentation du nombre de séquences

- Maintenant on va fixer le nombre de séquences et varié la taille des séquences



	Longueur de séquence	Temps d'exécution (s)
0	100	0.468381
1	150	0.806370
2	200	1.378557
3	250	2.175323
4	300	4.162269

Conclusion

Ce projet s'est concentré sur l'implémentation et l'analyse de l'algorithme d'alignement de séquences basé sur l'approche progressive en utilisant des profils. Voici un récapitulatif des principaux aspects et résultats obtenus au cours de ce projet :

- **Introduction à l'alignement de séquences :**
  Nous avons commencé par expliquer le principe de l'algorithme de Needleman et Wunsch pour l'alignement de séquences globales. Cet algorithme utilise une matrice de score et une phase de traceback pour obtenir l'alignement optimal de deux séquences.
  Des exemples d'alignements ont été fournis pour illustrer le fonctionnement de l'algorithme.
- **Motivation et applications :**

  L'alignement de séquences est une méthode essentielle en bioinformatique, utilisée pour identifier des régions homologues entre des séquences biologiques, prédire des structures de protéines, et étudier des relations évolutives entre des organismes.

- **Alignement multiple de séquences :**

  L'alignement multiple de séquences (MSA) étend le concept d'alignement de paires à plusieurs séquences simultanément, ce qui est crucial pour identifier des motifs conservés et des relations évolutives complexes.

  Nous avons implémenté l'approche progressive pour le MSA, en utilisant une matrice de distance et en alignant les séquences progressivement selon un arbre guide.

- **Tests et analyses :**

  Plusieurs tests ont été effectués pour évaluer les performances de l'algorithme d'alignement progressif. Nous avons testé l'algorithme sur différentes tailles de séquences (L = 50, 100, 150, 200, 250, 300) et pour des ensembles de séquences variés (3 à 10 séquences).

  Les résultats montrent que le temps d'exécution augmente avec la longueur des séquences et le nombre de séquences à aligner, ce qui est attendu étant donné la complexité croissante des calculs nécessaires pour gérer des profils de plus en plus grands.

- **Visualisation des résultats :**

  Les alignements obtenus ont été affichés avec des noms attribués aux séquences, permettant une lecture plus claire et une meilleure compréhension des résultats.

  Des graphiques ont été tracés pour illustrer l'impact de la longueur des séquences sur le temps d'exécution de l'algorithme, offrant une visualisation intuitive des performances de l'algorithme.

Perspectives et améliorations

Ce projet ouvre la voie à plusieurs améliorations et recherches futures :

- **Optimisation des algorithmes :** Des techniques avancées, telles que la parallélisation et les algorithmes heuristiques, pourraient être explorées pour améliorer les performances et réduire les temps d'exécution.
- **Extension aux séquences protéiques :** L'implémentation pourrait être étendue pour gérer les séquences de protéines, en adaptant les scores de substitution pour mieux refléter les propriétés biologiques des acides aminés.
- **Applications réelles :** L'algorithme pourrait être appliqué à des ensembles de données biologiques réels pour identifier des motifs conservés et des régions fonctionnelles importantes, contribuant ainsi à la recherche en génomique et en biologie moléculaire.

En conclusion, ce projet a permis de mettre en œuvre et d'analyser une méthode d'alignement multiple de séquences, démontrant son utilité et ses applications en bioinformatique. Les résultats obtenus offrent une base solide pour des travaux futurs visant à améliorer et à étendre les capacités de cette approche fondamentale.

Annexe