Bioinformatique

Travail d'étude

Damien Imbs et Mohamed Sayed Hassan

Université de Nice Sophia Antipolis

Table des matières

Chapitre 1

Introduction

L'étude du génome humain connait actuellement un énorme regain d'attention de la part du public, même si le nom de «génomique» est peu connu. Le séquençage complet du génome humain, relativement récent, a fait la une des journaux.

En effet, les gènes sont à la base de toute vie terrestre. Il est donc normal qu'en tant qu'êtres vivants, nous soyons fascinés par cette sorte de «programme» qui nous construit. Il est tout aussi normal que sa modification pose des problèmes éthiques profonds et sucsite des oppositions farouches.

Si la bioinformatique est très liée à l'étude des génomes, elle ne s'y limite pas. Elle concerne aussi l'étude des protéines, «briques» de base du vivant dans son ensemble. Elle connait également des développements moins directement liés au vivant ; la programmation biologique, utilisation d'ADN pour résoudre des problèmes informatiques, en est un bon exemple.

Nous allons donc présenter ces différents aspects de la bioinformatique.

Chapitre 2

Définition

2.1 Histoire du terme «bioinformatique»

Le terme de «bioinformatique» date du début des années 80. Cependant, le concept sous-jacent de traitement de l'information biologique est bien plus vieux. Durant les années 60, la biologie moléculaire a eu besoin de modélisation formelle, ce qui a mené à la création des «biomathématiques».

L'apparition de la bioinformatique n'est donc pas une conséquence de la génomique (séquençage d'un génome et son interprétation), mais plutôt une de ses fondations.

2.2 Buts

La bioinformatique est l'étude de l'information biologique. Ce n'est pas simplement l'application à la biologie de l'informatique ; c'est une branche à part entière de la biologie. La bioinformatique actuelle se concentre surtout sur l'étude des séquences d'ADN et sur le repliement des protéines, donc travaille surtout au niveau moléculaire. De nombreux bioinformaticiens travaillent également à l'élaboration d'outils biologiques permettant de résoudre des problèmes de l'informatique classique.

2.3 Apports à la biologie

L'informatique est devenue un apport fondamentale à la biologie moléculaire. Les moyens informatiques sont naturellement utilisés pour le stockage ou la gestion des données mais egalement pour l'interprétation de ces données. Le traitement informatique des séquences peut par exemple déterminer la fonction biologique d'un gène. Cet apport informatique concerne principalement quatre aspects :

- Le premier est l'organisation des données avec essentiellement la création de bases de données afin de réunir le plus d'information possible sur les séquences.

- Le deuxième aspect concerne les traitements que l'on peut effectuer sur les séquences afin de repérer un élément biologique intéressant. Ces programmes répresentent les traitements couramment utilisés dans l'analyse des séquences comme la recherche des similitudes d'une séquence avec l'ensemble d'une base de données.

- Le troisième aspect est celui qui permet d'élaborer des stratégies pour apporter des connaissances biologiques supplémentaires que l'on pourra ensuite intégrer dans des traitements standards. Par exemple la mise au point de nouvelles matrices de substitution des acides aminés, etc...

- Enfin, le quatrième aspect est celui de l'évaluation des différentes approches citées précédemment dans le but de valider.

Chapitre 3

Analyse de séquences d'ADN

3.1 Qu'est-ce qu'une séquence d'ADN ?

Une séquence génomique est l'enchaînement des nucléotides le long d'une macromolécule d'ADN. Elle peut être représentée par une chaîne de caractères utilisant l'alphabet A,C,G et T, initiales des bases azotées : Adénine, Cytosine, Guanine et Thymine.

3.1.1 Pourquoi comparer les séquences

La comparaison de séquences est la tâche informatique la plus utilisée par les biologistes. Il s'agit dans quelle mesure deux séquences, génomiques, se ressemblent. Ainsi, si deux séquences sont très similaires et si l'une est connue pour être codante, l'hypothèse que la seconde le soit aussi peut etre avancée. Un biologiste qui detient une nouvelle séquence s'intéresse en premier temps à parcourir ces bases de données, afin d'y trouver les séquences similaires et de faire hériter à la nouvelle séquence les connaissances qui leur sont associées. C'est également en comparant des séquences de génomes d'espèces actuelles qu'il est possible de reconstruire des arbres phylogénétiques qui rendent compte de l'histoire évolutive.

3.2 Bases de données spécifiques

3.2.1 Introduction

Il existe un grand nombre de bases de données d'intérêt biologique. Il y a deux sortes de banques :

- celles qui offrent des informations plutôt hétérogènes.
- celles qui correspondent à des données plus homogènes d'espèces précises.

Il est fréquent d'appeler les premières 'banques de données' et les secondes 'bases de données', mais cette distinction n'est pas très connue par les non-biologistes. Pour éviter toute confusion, nous appellerons les premières banques de données ou bases de données généralistes et les secondes spécialisées.

3.2.2 Les banques généralistes

Genbank (banque américaine créée en 1982) et EMBL (banque européenne qui existe depuis 1980) sont les grandes banques de séquences généralistes. Leur mission est de rendre publiques les séquences qui ont été déterminées. On trouve également une expertise biologique directement liées aux séquences traitées.

3.2.3 Les banques spécialisées

De nombreuses bases de données spécifiques ont été créées pour des besoins spécifiques liés à l'activité d'un groupe de personnes. Elles ont pour but de recenser des familles de séquences autour de caractéristiques biologiques comme les gènes identiques issus d'espèces différentes. Elles peuvent aussi régrouper des classes spécifiques de séquences comme les vecteurs de clonage ou toutes les séquences d'un même génome.

3.3 Diffusion et utilisation des banques de données

3.3.1 La diffusion

Les bases de données sont mises à jour plusieurs fois par an. Pendant longtemps, le mode de distribution etait l'envoi postal. Depuis 1990 et le développement des réseaux informatiques à haut débit, un grand nombre de base sont stockés sur des serveurs publics. Ainsi beaucoup de serveurs mettent à disposition de nombreuses bases avec une mise à jour quotidienne des données (comme la banque EMBL).

3.3.2 Interrogation

Il existe deux types de logiciels pour que les utilisateurs puissent extraire les informations qui les intéressent. Les premiers sont des logiciels déjà programmés (comme le logiciel ACNUC ou SRS). Par contre, les deuxièmes sont des programmes établis à l'aide de systèmes de gestion de bases de données (SGBD) qui utilise un langage de requête.

3.4 Les Algorithmes et les programmes de comparaison de séquences

3.4.1 Algorithme de recherche de similitudes

La première étape des analyses de séquences est la recherche de similitude entre séquences. Elle permet de révéler des régions proches dans leur séquence primaire en considérant le minimum de changements en insertion, suppression, ou substitution qui séparent deux séquences. Ainsi des informations importantes seront reperées sur la structure, la fonction ou l'évolution des biomolécules. Cette méthode est largement utilisée dans la caractérisation de régions communes ou similaires entre deux ou plusieurs séquences, dans la comparaison d'une séquence avec l'ensemble ou sous-ensemble des séquences d'une base de données. Le problème de la comparaison de séquences est que les séquences à comparer ont rarement la même longueur et même si elles ont la même longueur rien ne dit qu'elles doivent être comparées sur cette longueur exactement. L'algorithme utilisé est basé sur la comparaison de fenêtres de longueur fixe que l'on déplace le long des séquences. Soit deux séquences A et B de longueurs respectives m et n à comparer et l la longueur de la fenêtre. Soit f la première fenêtre de longueur l sur la séquence A. Soit i une indice à incrémenter pour déterminer la prochaine fenêtre à prendre. On va comparer f avec toutes les fenêtres possibles de même longueur, obtenues à partir de la séquence B. On incrémente i. Puis on recommence les comparaisons sur la séquence B. Si on incrémente i de 1, on effectuera $O(n \times m)$ comparaisons de fenêtres différentes. Pour chaque comparaison entre deux fenêtres, on attribue un score. On mémorisera uniquement les comparaisons dont les scores sont supérieurs ou égaux à un seuil fixé précédemment (une application directe de cet algorithme est le programme de Diagon de Staden. Ce programme dessine graphiquement les points de similitude entre deux séquences).

3.4.2 Algorithme de la recherche de segments identiques

L'objectif est de retrouver les zones identiques entre deux séquences. L'une des algorithmes les plus utiliséss est celle de Dumas et Ninio (1982). Elle permet la transformation d'une séquence en suite d'entiers à partir de la description faite en chaîne de caractères. Pour cela, l'algo consiste à composer une séquence en autant de segments de longueur fixe. Puis elle donne un code à chacun de ces segments. Le code est un entier attribué en fonction de l'alphabet(A,C,T ou G). Pour détecter les ségments identiques il suffit de réperer les codes indentiques. La rapidité de la méthode est proportionnelle à la longueur du mot codé, et plus cette longueur est grande, plus le résultat est grossier.

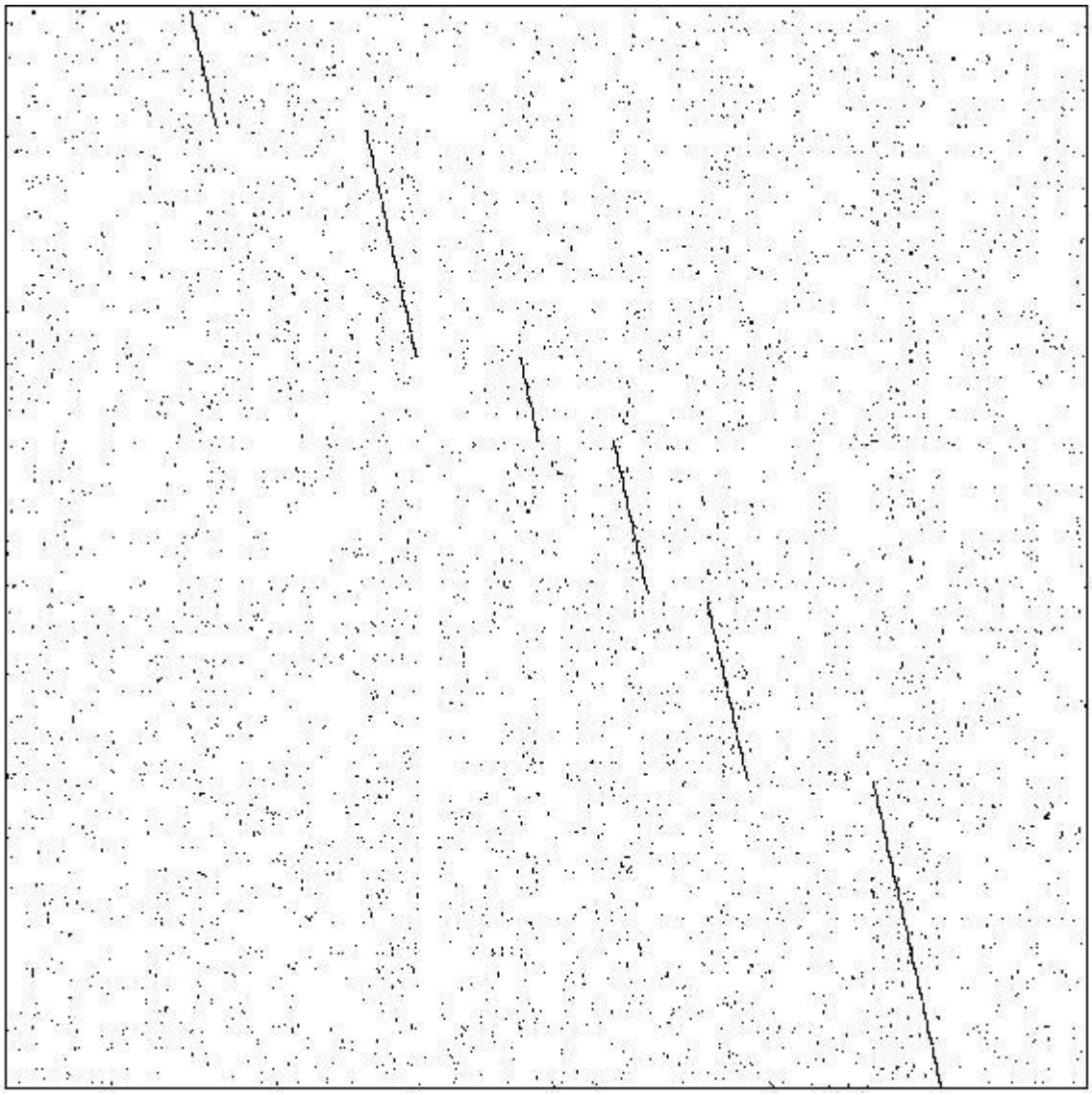


FIG. 3.1 – Comparaison de deux séquences d'ADN

3.4.3 Algorithme d'alignement optimal

La plupart des algorithmes utilisés sont basés sur la programmaton dynamique. L'algorithme de Needleman-Wunsch fournit le meilleur alignement global entre deux séquences (Needleman et Wunsch 1970). On parle d'alignement global car elle prend en compte tous les éléments de chacune des séquences. Le principe de l'algorithme consiste à calculer les scores maximaux d'alignements entre tous les préfixes de U et de V de deux séquences. Afin d'optimiser la comparaison de deux séquences, on peut introduire des insertions ou des délétions de longueur variables à des positions différentes. On parle de gap (figure 3.2).

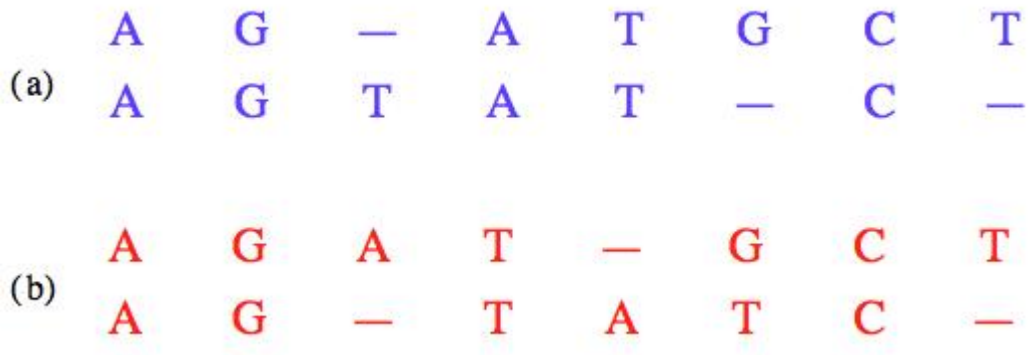


FIG. 3.2 – Deux alignements possibles de la meme paire de séquences avec insertion de gap.

Soit M la matrice avec ou chaque colonne correspond à une lettre de la première séquence et chaque ligne à une lettre de la seconde. Notons s(a,b) le score obtenu en alignant le caractère a avec la caractère b. Soit $M_{i,j}$ le score optimal de l'alignement entre $U_1 \ldots U_i$ et $V_1 \ldots V_j$ et d le cout d'insertion ou délétion. Si on veut calculer un alignement optimal entre $U_1 \ldots U_i$ et $V_1 \ldots V_j$ alors on sait que ça provient de l'un des 3 cas ci-dessous

Il suffit de prendre le meilleur de trois alignement pour avoir un coût minimal. D'où les formules :

- $M(0,0) = 0$
- $M(i,0) = \sum_{k=1}^{i} s(U_{k,})$
- $M(0,j) = \sum_{k=1}^{j} s(,V_k)$
- $M(i,j) = Max(M(i-1,j-1)+s(U_i,V_j), M(i-1,j)+s(U_i,gap), M(i,j-1,s(gap,V_j))$

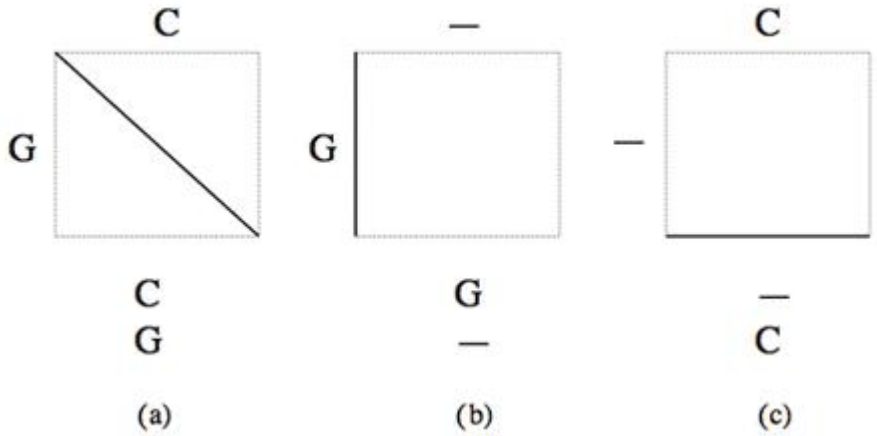


FIG. 3.3 – (a) un trait en diagonale indique la mise en correspondance d'un caractère d'une séquence avec un caractère de l'autre, un trait vertical (b) indique la mise en correspondance d'un caractère de la première séquence avec un gap inséré dans la seconde, et un trait horizontal (c) l'insertion d'un gap dans la première séquence en face d'un caractère de la seconde

La complexité de cet algorithme est $O(m \times n)$ en temps et en espace. Pour un exemple de représentation des alignements entre 2 séquences, voir la figure 3.4).

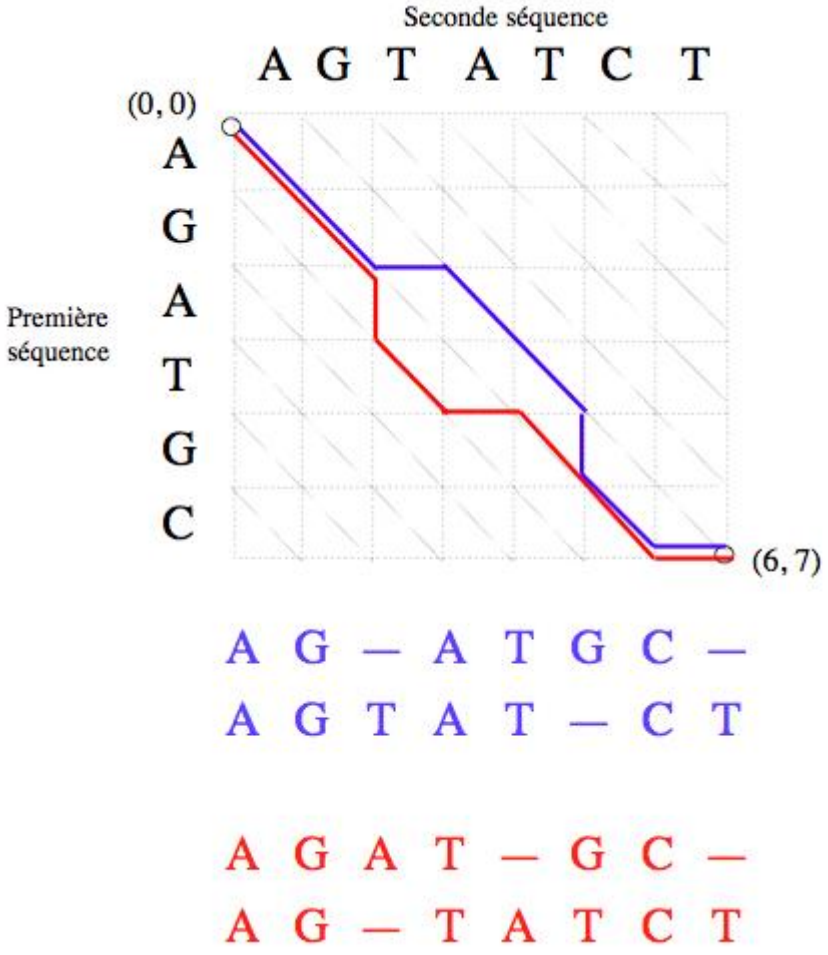


FIG. 3.4 – Représentation graphique des alignements.

Chapitre 4

Analyse de la structure de protéines

4.1 Qu'est-ce qu'une protéine ?

Les protéines sont des longues chaînes d'acides aminés. Elles sont à la base de l'activité biologique au niveau microscopique. Les enzymes, les anticorps, ainsi que toutes les autres structures complexes du vivant sont composés de protéines.

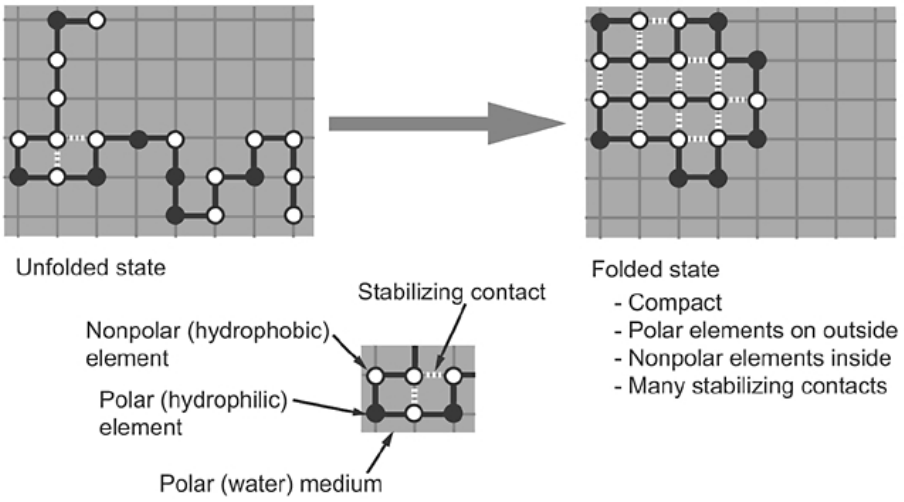
Il y a 20 acides aminés différents présents dans les protéines, codés chacun par une séquence de 3 bases d'ADN (4 bases possibles : A, C, G et T pour l'ADN ; A, C, G et U pour l'ARN). Comme $4^3 = 64$, chaque acide aminé est codé par plusieurs séquences de bases. Il y a aussi des séquences pour coder la fin d'une protéine. Voir figure 4.1 page 17.

Une fois la protéine produite, elle se replie. L'étude de ce repliement est la principale préoccupation de la bioinformatique structurale. En effet, connaître la séquence d'acides aminés composant une protéine ne suffit pas pour connaître sa fonction. La fonction d'une protéine dépend de la structure qu'elle adopte en se repliant.

4.2 Quels mécanismes forcent une protéine à se replier ?

Comme nous l'avons dit, les protéines sont des longues chaînes d'acides aminés. Certains de ces acides aminés sont hydrophiles, d'autres sont hydrophobes. À une température suffisante, les parties hydrophobes ont généralement tendance à se replier vers le centre de la protéine, là où elles sont le moins exposées à l'eau environnante. Ce n'est pas la seule force qui entre en jeu. Entre autres, les charges électriques jouent aussi un rôle. Finalement, la protéine est stabilisée par des liaisons entre acides aminés.

Examinons l'exemple (très simplifié) suivant :



Les éléments hydrophobes (en blanc dans l'exemple) se regroupent pour fuir l'eau environnante, alors que les éléments hydrophiles restent à l'extérieur. Enfin, des liaisons se créent entre les acides aminés ; cela stabilise la protéine.

Il est intéressant de noter que généralement, ces liaisons ne sont pas définitives. La protéine peut être dépliée dans certaines conditions (entre autres en abaissant la température).

Ce qui permet à ces forces de plier la protéine est la relative liberté de mouvement que permettent les liens entre les acides aminés. Ces liens ne sont cependant pas complètement libres et influencent aussi la structure finale de la protéine.

4.3 Enjeux

- Les maladies génétiques :
  Si une protéine ne se replie pas correctement (généralement à cause d'une mutation qui a changé sa séquence d'acides aminés), elle ne pourra pas assurer sa fonction. C'est le problème des maladies génétiques. Comprendre la structure et la fonction de la protéine en cause permet ensuite d'élaborer des thérapies adaptées, en particulier des thérapies géniques.

- Les nanotechnologies :
  Les protéines et leur production par les organismes vivants sont une

merveille de précision à l'échelle microscopique. On essaie actuellement de produire des nanomachines à base de protéines.

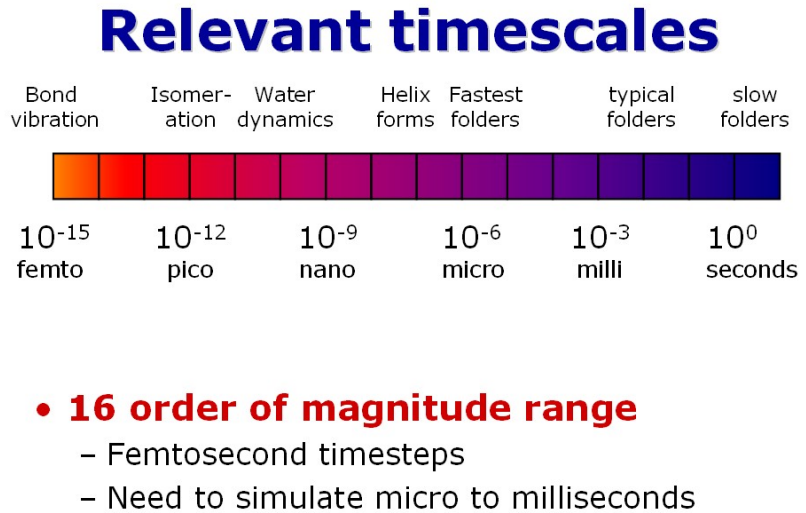
4.4 Détermination de la structure d'une protéine à partir de la séquence d'acides aminés correspondante

4.4.1 Problématique

On ne peut pas connaître directement la structure d'une protéine (et donc sa fonction) avec sa séquence d'acides aminés. Il faut donc étudier en détail la façon dont elle se replie. Tout le problème vient de la complexité de ce repliement.

A l'échelle humaine, les protéines se replient très rapidement : de l'ordre d'une microseconde pour les plus rapides, une seconde pour les plus lentes.

Mais ce repliement se fait en fonction d'événements à une échelle de temps encore plus réduite : les vibrations des liens entre les acides aminés, qui sont à la base du repliement des protéines, sont de l'ordre d'une femtoseconde (10^{-15} seconde). C'est donc à cette échelle qu'il faut travailler si on veut modéliser le repliement d'une protéine. Cela nécessite une puissance de calcul gigantesque.



Pour faire face à cette complexité, on distingue deux approches principales : les modèles état-espace (Discrete State-space Models, ou DSMs) et la force brute (la plus récente).

4.4.2 L'approche «brutale»

C'est l'approche la plus récente. Elle est utilisée par le projet folding@home [4]. Elle utilise un cluster de centaines de milliers d'ordinateurs, en utilisant les CPU de volontaires à travers le monde. Ce projet a été inspiré par le projet seti@home.

Elle consiste à lancer en parallèle des milliers de simulations (chacune portant sur une protéine) chargées d'étudier une partie du repliement de la protéine. Les données récoltées sont ensuite unifiées et stockées dans la «Protein Data Bank» (une base de donnée qui est une référence mondiale pour les structures de protéines).

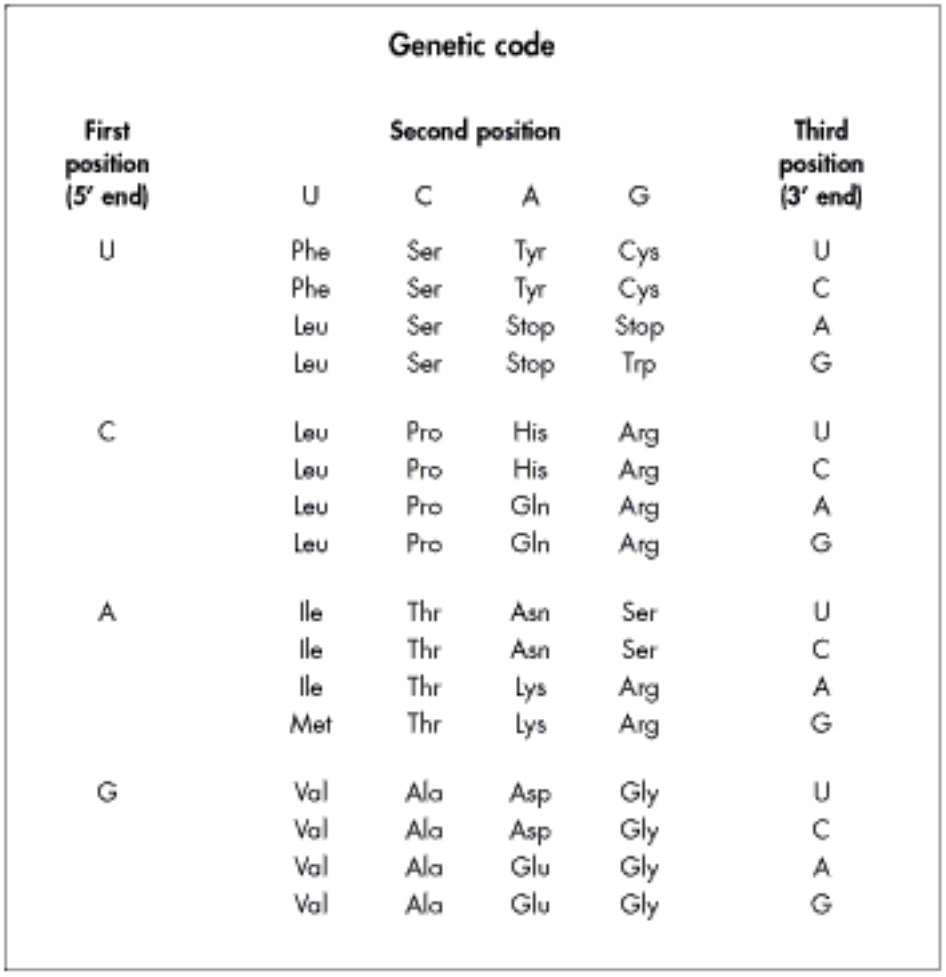
4.4.3 L'approche DSMs (Discrete State-space Models)

C'est l'approche utilisée, entre autres, par le projet PSA (Protein Sequence Analysis) de l'université de Boston [6]. Elle utilise des modèles de Markov cachés (*Hidden Markov Models* ou *HMMs*, pour plus de détails voir [2]). Elle est décrite par [5].

L'idée générale est de comparer différents modèles, appelés «Discrete State-space models», et de voir lequel a la plus grande probabilité de correspondre à la séquence.

Dans ces modèles, le facteur temps représente la position de chaque acide aminé dans la séquence. Les différents états représentent les 20 acides aminés. Les probabilités de transitions entre les états représentent la probabilité que l'acide aminé suivant corresponde au modèle.

Cette méthode permet une classification bien plus rapide des protéines, mais ne permet pas la précision de l'approche «brutale». En effet, il y a une marge d'erreur. On la réduit généralement en comparant la séquence à celles de protéines déjà connues.



Genetic code

First position (5′ end)	Second position U	C	A	G	Third position (3′ end)
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	Stop	Stop	A
	Leu	Ser	Stop	Trp	G
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Met	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Glu	Gly	G

FIG. 4.1 – Le codage des acides aminés

Chapitre 5

Programmation biologique

5.1 Présentation

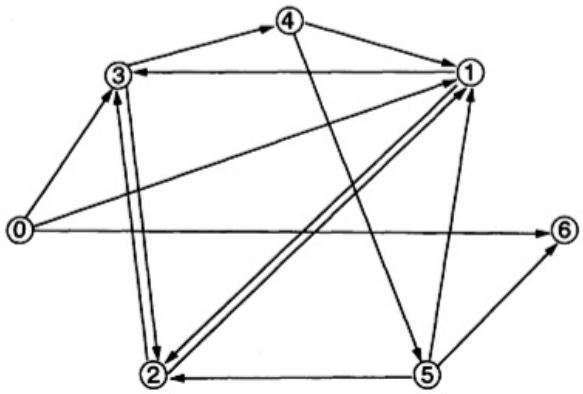
On appelle programmation biologique, ou calcul biologique, la résolution de problèmes par des moyens biologiques. Pour certains problèmes, la puissance d'un ordinateur biologique est théoriquement très superieure à celle d'un ordinateur classique [1]. En effet, les techniques issues de la biologie permettent de faire des calculs massivement parallèles.

5.2 Exemples simples

5.2.1 Problème du chemin hamiltonien

Le premier résultat experimental a été obtenu par Adleman, qui a résolu par des moyens purement biologiques le problème de l'existence d'un chemin hamiltonien dans un graphe.

Dans son article de 1994 [1], il décrit la méthode qu'il a utilisé pour déterminer l'existence d'un chemin hamiltonien dans un graphe à 7 sommets.



Il a appliqué l'algorithme non-déterministe suivant :

1. Générer des chemins aléatoires à travers le graphe.
2. Garder seulement ceux qui partent du sommet initial et qui arrivent au sommet final.
3. Si le graphe a n sommets, ne garder que les chemins de longueur $n-1$.
4. Ne garder que les chemins qui passent à travers tous les sommets.
5. Si il reste des chemins, répondre «vrai». Sinon, répondre «faux».

Pour arriver à ce but, il a créé des séquences d'ADN de 20 bases chacunes, une pour chaque sommet et pour chaque arète. Pour les sommets, la séquence était choisie au hasard. Pour les arètes partant du sommet initial, la séquence était la totalité de ce sommet, plus les 10 premières bases du sommet d'arrivée. Pour les arètes arrivant au sommet final, elle était les 10 dernières bases du sommet de départ, plus la totalité du sommet d'arrivée. Pour les autres arètes, les 10 premières bases était les 10 dernières du sommet de départ, les 10 dernières était les 10 premières du sommet d'arrivée.

Il a ensuite mélangé une quantité suffisante des séquences correspondant aux arètes avec des séquences complémentaires à celles des sommets. Cela a permis de générer des chemins aléatoires à travers le graphe (étape 1).

Ensuite, grâce à des procédés chimiques que nous ne détaillerons pas, il a isolé les séquences correspondant aux chemins partant du sommet initial et arrivant au sommet final (étape 2). Cela a été possible car ces chemins étaient les seuls à contenir les séquences des sommets initial et final.

Il a isolé les chemins de longueurs $n-1$ (étape 3). Cela a été possible en isolant les séquences de bonne longueur.

Pour chaque sommet, il n'a gardé que les chemins passant par ce sommet (étape 4). Pour cela, il a isolé les séquences contenant la séquence du sommet.

Il a ensuite cherché si il restait des chemins, c'est-à-dire des séquences (étape 5). Dans son cas, c'était vrai ; il avait résolu le problème.

Il faut remarquer que la quantité de séquences utilisée doit être suffisante pour qu'il y ai une très forte probabilité qu'un chemin se forme.

Ce problème est NP-complet ; grâce à cette méthode, il peut être résolu (relativement) simplement.

5.2.2 Problème 3SAT

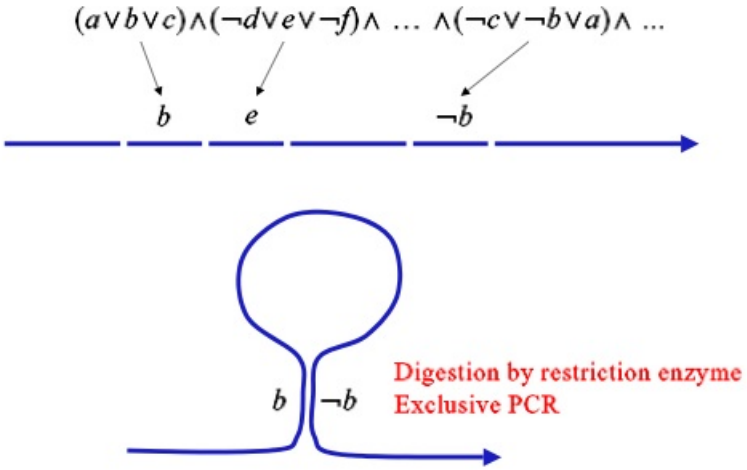
Voici un autre problème NP-complet. Dans ce problème de satisfiabilité, chaque clause contient exactement 3 littéraux.

Une technique à base d'ADN a également été développée pour le résoudre ([3], pages 5-6). Le principe peut cependant être généralisé au problème SAT général.

Pour chaque littéral(variable booléenne), on crée une séquence. La négation de ce littéral est représentée par la séquence complémentaire.

On crée ensuite un grand nombre de séquences contenant chacune un littéral de chaque clause. Ces séquences représentent les différents états pouvant satisfaire la formule. Si un état contient à la fois un littéral et sa négation, l'état n'est pas valide.

Si une séquence contient à la fois la représentation d'un littéral et celle de sa négation, elle forme une boucle comme illustré ci-dessous :



Grâce à un enzyme approprié, on élimine toutes les séquences contenant des boucles, c'est-à-dire toutes les représentations d'états invalides. Si il reste des séquences, donc des états valides, la formule est satisfiable.

Chapitre 6

Conclusion

La bioinformatique est un domaine de recherche très actif actuellement. L'abondance d'articles, de thèses, de conférence la concernant en est une preuve.

L'attention du public et l'interêt des gouvernements lui permettent pour l'instant de maintenir cette activité grâce à des financements conséquents.

Heureusement, car il reste beaucoup à découvrir dans ce domaine. C'est un domaine scientifique très jeune et qui n'a dévoilé pour l'instant qu'une partie infime de ses possibilités. L'avenir nous réserve de grandes surprises ; nous les attendons avec impatience.

Bibliographie

[1] Leonard M. Adleman. Molecular computation of solutions to combinatorial problems. *Science*, 266(5187) :1021–1024, November 1994.

[2] R. Boyle. Hidden markov models. `http://www.comp.leeds.ac.uk/roger/HiddenMarkovModels/html_dev/main.html`.

[3] Masami Hagiya. Towards molecular programming.

[4] University of Stanford. Folding@home website. `http://folding.stanford.edu`.

[5] Collin M. Stultz, James V. White, and Temple F. Smith. Structural analysis based on state-space modeling. *Protein Science*, 2 :305–314, 1993.

[6] Boston University. The psa structure prediction server. `http://bmerc-www.bu.edu/psa/`.