UFR de Biologie
2 place Jussieu
Tour 42
75005 Paris

Unité Mathématique
Informatique et Génomes - INRA
Domaine de Vilvert
78352 Jouy en Josas Cedex

UNIVERSITE PARIS VII - DENIS DIDEROT

Ecole Doctorale B2M

THESE

pour l'obtention du diplôme de

DOCTEUR DE L'UNIVERSITE PARIS VII

Spécialité : Analyse des Génomes et Modélisation Moléculaire

par

Juliette Martin

Prédiction de la structure locale des protéines par des modèles de chaînes de Markov cachées

Thèse soutenue publiquement le 17 novembre 2005 devant le jury :

Pr. Catherine Etchebest, Université Paris 7	Président
Pr. Gilbert Deléage, Université Lyon 1	Rapporteur
Dr. Jean-Loup Risler, CNRS	Rapporteur
Dr. Anne-Claude Camproux, Université Paris 7	Examinateur
Dr. Joël Pothier, Université Paris 6	Examinateur
Dr. Jean-François Gibrat, INRA	Directeur de thèse
Dr. François Rodolphe, INRA	Directeur de thèse

Remerciements

Je remercie Jean-François Gibrat et François Rodolphe qui ont assuré la direction “à deux têtes” de ce travail de thèse.

Merci à Gilbert Deléage et Jean-Loup Risler pour avoir accepté de juger mon travail, à Catherine Etchebest qui a bien voulu présider ce jury ainsi qu'à Anne-Claude Camproux et Joël Pothier pour l'intérêt qu'ils ont porté à mon travail en tant qu'examinateurs.

Un grand merci à mes relecteurs de choc : Alexandre de Brevern, Florence Muri, Jean-François Taly, Pierre Nicolas et Etienne Roquain.

Je tiens à remercier les stagiaires dont le travail m'a permis d'avancer : Guillaume Letellier, Emeline Legros et Claire Guillet.

Merci à tous et à toutes les MIG-ales qui font de MIG un lieu de travail à l'ambiance sympathique et décontractée (et en plus on rafle toutes les coupes à l'inter-labo ! !). Mention spéciale à Jeff pour les blagues et les gâteaux quand ça n'allait pas, et le “coaching” un peu tout le temps, et à Pierre pour le soutien psychologique, notamment dans la dernière ligne droite.

Merci aux participants du groupe SSB, qui contribuent à ma familiarisation avec les statistiques, en particulier Pierre-Yves Bourguignon avec qui j'ai eu le plaisir de travailler.

Merci enfin à Julien pour son soutien, son écoute et sa patience à toute épreuve tout au long de ces trois années. Il en a appris bien plus qu'il n'en demandait sur le monde magique des HMM, tout en continuant à avancer avec succès dans son domaine.

Sommaire

Introduction **1**

1 Les protéines : Niveaux d'organisation et méthodes de prédiction **5**

1.1 Structure des protéines 7

1.1.1 Structure primaire 7

1.1.2 Structure secondaire 11

1.1.3 Structure tertiaire 14

1.1.4 Structure quaternaire 15

1.2 Prédiction de la structure des protéines 16

1.2.1 Prédiction de la structure locale des protéines 16

1.2.2 Prédiction de la structure globale des protéines 27

1.3 Conclusion 33

2 Une nouvelle méthode d'assignation des structures secondaires des protéines **35**

2.1 Contexte 36

2.2 Matériel et méthodes 42

2.2.1 Données 42

2.2.2 Nouvelle méthode d'assignation : KAKSI 43

2.2.3 Méthodes d'assignation utilisées pour la comparaison 48

2.2.4 Indices globaux de comparaison 49

2.2.5 Analyse de la géométrie des hélices avec un logiciel externe 50

2.3 Résultats et discussion 51

2.3.1 Paramètres internes de KAKSI 51

2.3.2 Sensibilité des méthodes envers la résolution 53

2.3.3 Mesure de l'accord global entre les méthodes 54
2.3.4 Comparaison détaillée 59
2.4 Quelques exemples de structures analysées différemment par les programmes d'assignation 67
2.5 Conclusion 70

3 Les HMM et leur utilisation en modélisation mathématique des protéines 71
3.1 Présentation des modèles HMM 72
3.1.1 Spécification du modèle 73
3.1.2 Algorithmes de prédiction 77
3.1.3 Estimation des paramètres 83
3.1.4 Quelques généralisations des modèles de Markov 87
3.2 Applications de HMM à la prédiction de structure des protéines : état de l'art 91
3.2.1 Prédiction de la structure globale des protéines par les profils HMM 91
3.2.2 Prédiction de la structure locale des protéines par les HMM 93
3.3 Conclusion 107

4 Choix de modèles pour la prédiction de structure locale des protéines 109
4.1 Matériel et méthodes 111
4.1.1 Données 111
4.1.2 Utilisation des HMM 112
4.1.3 Indices de prédiction 112
4.2 Modèles à trois états cachés pour la prédiction de structure secondaire . . . 114
4.2.1 Modèles M1Mn 114
4.2.2 Modèles MTD 116
4.2.3 Modèles "à trous" 119
4.2.4 Modèles parcimonieux 120
4.2.5 Conclusion 122
4.3 Proposition de modèle M1M0 construit avec des *a priori* biologiques 123
4.3.1 Modèle d'hélice α 123
4.3.2 Modèle de brin β 124

4.3.3 Proposition d'un modèle complet pour les structures secondaires . . 126
4.3.4 Estimation des paramètres et utilisation du modèle 127
4.4 Choix d'un modèle M1M0 parmi des modèles générés automatiquement, sans *a priori* 129
4.4.1 Estimation des paramètres et utilisation des modèles 130
4.4.2 Modèles ayant le même nombre d'états cachés par classe 130
4.4.3 Modèles ayant un nombre d'états cachés différent dans chaque classe 135
4.4.4 Performances du modèle optimal pour la prédiction de structures secondaires 144
4.4.5 Score de confiance de la prédiction 145
4.4.6 Conclusion 146
4.5 Prédiction des angles dièdres à l'aide des HMM 146
4.5.1 Zones d'angles dièdres 146
4.5.2 Modèle pour les zones d'angles 147
4.5.3 Prédiction des zones d'angles 147
4.6 Conclusion 148

5 Tentative d'intégration d'une information à longue portée dans les modèles 151
5.1 Contexte et objectif 152
5.2 Propositions de fonctions de score d'appariement 153
5.2.1 Fonction de score basée sur les paires de résidus face à face dans les brins β appariés en feuillets β 153
5.2.2 Fonction de score basée sur les probabilités de triplets appariés dans les brins β 158
5.3 Utilisation directe de la fonction de score dans la prédiction par le HMM . 160
5.4 Utilisation *a posteriori* de la fonction de score pour modifier la prédiction fournie par le HMM 162
5.5 Discussion 163
5.6 Conclusion 165

6 Utilisation des séquences homologues dans la prédiction 167
6.1 Contexte et objectif 168

6.2 Matériel et méthodes 170
6.2.1 Données 170
6.2.2 Systèmes de pondération des séquences 171
6.2.3 Pondération de Henikoff et Henikoff, 1994 [81] 171
6.2.4 Pondération basée sur les arbres phylogénétiques, Thompson et al, 1994 [190] 172
6.2.5 Modèle d'évolution 173
6.2.6 Pondération équitable utilisant l'arbre phylogénétique 174
6.2.7 Prise en compte directe de l'arbre phylogénétique dans la prédiction par forward/backward 181
6.2.8 Simulation de données à l'aide d'un HMM et d'un arbre 183
6.3 Résultats 184
6.3.1 Résultats sur séquences réelles 184
6.3.2 Résultats sur séquences simulées 187
6.4 Conclusion 192

Conclusion et perspectives **193**

Bibliograhie **197**

Annexes **214**
A1 Méthode Rosetta 216
A2 Loi stationnaire d'une chaîne de Markov 220
A3 Complément pour le calcul de l'information portée par un arbre phylogénétique 222
A4 Articles 224

Introduction

Les protéines sont des acteurs majeurs du monde vivant. Elles sont un constituant des cellules et jouent un rôle crucial dans les processus biologiques nécessaires au fonctionnement des cellules. Une protéine est une macromolécule complexe, constituée d'un enchaînement séquentiel d'acides aminés, qui s'organise dans l'espace en une structure tridimensionnelle (3D) particulière. La fonction d'une protéine dépend de manière critique de sa structure 3D. Aussi, la connaissance de la structure 3D est indispensable à la compréhension atomique de l'activité biologique.

La détermination expérimentale de la structure 3D est un processus beaucoup plus long et coûteux que le sequençage des génomes qui permet d'obtenir rapidement un très grand nombre de séquences protéiques. Cet écart entre la masse de données séquentielles et la masse de données structurales a conduit au développement de méthodes dont le but est de prédire *in silico* la structure des protéines.

La structure 3D d'une protéine est dictée par sa séquence en acides aminés. Les méthodes de prédiction *in silico* utilisent cette propriété pour prédire la structure 3D d'après la séquence. Les techniques mises en oeuvre dépendent des informations disponibles sur la protéine à prédire. Dans le cas favorable, il sera possible d'utiliser des structures déjà déterminées pour dériver un modèle global de la protéine à prédire. Cette approche se fonde sur la notion d'homologie : des protéines issues d'un ancêtre commun conservent des structures 3D similaires, malgré les modifications des séquences dues aux mutations. Dans le cas le moins favorable, il n'est pas possible de dériver un modèle global à partir des structures disponibles. Les méthodes visant à fournir de prédictions dans ce dernier cas sont dites *de novo*.

La prédiction de structure *de novo* peut être attaquée de front. Un modèle physique simplifié de la séquence protéique est élaboré, et la structure la plus stable est recherchée d'après une fonction d'énergie empirique. Les limitations intrinsèques de cette approche ne permettent pas de fournir des prédictions pour des protéines de taille importante. Une autre approche, indirecte, consiste à découper la séquence en fragments dont les structures sont prédites séparément, puis assemblées. Ce type d'approche est appelé *approche par fragments* (*fragment-based approach*). Le succès récent des approches par fragments nous a incité à suivre cette voie.

La prédiction de structure *de novo* par fragments nécessite une étape de prédiction de structure locale, afin de modéliser les fragments à assembler. Dans le cadre du déve-

loppement d'une telle méthode, ma thèse a donc porté sur la prédiction de la structure locale des protéines d'après leur séquence. La méthodologie utilisée pour la prédiction de structure locale repose sur l'utilisation de modèles statistiques appelés chaînes de Markov cachées ou HMM (Hidden Markov Model).

Le premier chapitre de cette thèse présente le cadre biologique de l'étude : les différents niveaux de la structure des protéines, ainsi qu'un état de l'art des méthodes existantes de prédiction de structure *in silico*. Le deuxième chapitre propose une nouvelle méthode d'assignation des éléments de structures secondaires afin d'obtenir une description satisfaisante des structures 3D. Une étude comparative est présentée entre les assignations fournies par notre méthode et celles fournies par les programmes existants. Le troisième chapitre présente le cadre mathématique utilisé pour la prédiction locale : modèles de chaînes de Markov cachées et état de l'art de l'utilisation de ces modèles en prédiction de structure *in silico*. Le quatrième chapitre porte sur le choix des modèles de chaînes de Markov cachées pour prédire les structures secondaires et les angles dièdres de la chaîne principale des protéines. Des modèles à trois états cachés utilisant des schémas de dépendance variés sont envisagés. La recherche s'oriente ensuite vers des modèles incorporant les *a priori* biologiques sur les séquences. Puis des modèles sans *a priori* sont sélectionnés sur des critères objectifs et statistiques. Le cinquième chapitre rapporte un essai, d'introduction de contrainte à longue porté dans la méthode, visant à améliorer la prédiction des brins β. Le sixième et dernier chapitre concerne l'utilisation des séquences homologues pour améliorer la prédiction de structure locale par les modèles de chaînes de Markov cachées. Les séquences homologues sont utilisées grâce à des systèmes de pondérations des séquences pour combiner les prédictions menées sur les séquences d'une famille, ou par un couplage direct du HMM avec la phylogénie des séquences.

Chapitre 1

Les protéines : Niveaux d'organisation et méthodes de prédiction

Les protéines sont des macromolécules biologiques qui jouent un rôle essentiel dans tous les organismes vivants. En effet, elles sont partie intégrante de l'architecture des cellules et des tissus physiologiques. Elles sont aussi impliquées, à divers niveaux, dans le fonctionnement de la machinerie cellulaire. Ainsi, la plupart des enzymes qui catalysent les réactions biochimiques sont des protéines. Les protéines assurent également des fonctions de transport, par exemple dans les processus de signalisation cellulaire.

Une protéine est un long polymère, dont la séquence est codée par le matériel génétique des cellules (ADN), dans des portions codantes appelées gènes. Cette nature séquentielle des protéines a été mise en évidence pour la première fois par Frederick Sanger lors du séquençage de l'hormone insuline (prix Nobel de chimie en 1958).

Dans les conditions physiologiques, une séquence protéique ne reste pas sous forme d'un long filament non structuré. Elle s'organise dans l'espace en une structure tridimensionnelle la plupart du temps stable et bien déterminée. La structure tridimensionnelle (3D) d'une protéine lui permet d'exercer son activité biologique. Ainsi, une altération de la structure 3D entraîne une diminution, voir la perte de l'activité. La connaissance de la structure 3D est donc essentielle pour bien comprendre le fonctionnement d'une protéine.

Une séquence donnée de protéine mène à une structure fonctionnelle unique. Les expériences de Christian B. Anfinsen ont ainsi montré qu'une protéine dénaturée reprend spontanément sa conformation initiale, dite native, si la dénaturation n'est pas trop forte, lorsqu'elle est replacée dans ses conditions habituelles de milieu (prix Nobel de Chimie en 1972). On suppose que cette structure native correspond au minimum de l'énergie libre de la protéine. Théoriquement, il serait donc possible de prédire la structure tridimensionnelle d'une protéine d'après sa séquence.

Dans le contexte actuel, alors que les programmes de séquençage des génomes fournissent d'importantes quantités de données relatives aux séquences d'ADN -donc de protéines- l'enjeu est de taille : pour de très nombreuses protéines, seule la séquence est connue. La détermination expérimentale de la structure est une tâche beaucoup plus longue, difficile et coûteuse que le séquençage. La prédiction théorique des structures tridimensionnelles est donc devenue une nécessité pour compléter nos connaissances actuelles sur les génomes.

La structure 3D d'une protéine est une donnée complexe. Ce premier chapitre présente, dans une première partie, les différents niveaux de description des structures protéiques.

La deuxième partie présente les principales stratégies de prédiction de structure à partir des séquences de protéines.

1.1 Structure des protéines

La structure des protéines peut être abordée en utilisant la description hiérarchique suivante :

1. La structure primaire d'une protéine correspond à sa séquence, autrement dit sa formule chimique.
2. La structure secondaire est la description de sa structure locale en terme d'éléments répétés.
3. La structure tertiaire désigne la structure tridimensionnelle : la position de tous les atomes de la protéine dans l'espace.
4. La structure quaternaire décrit l'association de plusieurs protéines en un complexe, comme, par exemple, les protéines multimériques.

1.1.1 Structure primaire

Les protéines sont des polymères dont la brique élémentaire est l'acide aminé. Un acide aminé est une molécule composée d'un carbone asymétrique, le carbone α, lié à une fonction amine NH_2, une fonction carboxyle $COOH$, un hydrogène et une chaîne latérale ou radical (R). Il existe 20 acides aminés principaux, qui diffèrent par la nature de leur radical R. Les formules semi-développées des 20 acides aminés sont présentées dans la figure 1.1. La nature de ce radical confère aux acides aminés leurs propriétés physico-chimiques particulières.

Classification des acides aminés

Les acides aminés peuvent être regroupés sur la base de leurs propriétés physico-chimiques. D'après l'hydrophobicité (aversion pour l'eau) de leurs chaînes latérales, les acides aminés peuvent être qualifiés d'hydrophobes, polaires et chargés comme indiqué sur la figure 1.1. Cette classification donne un aperçu de la complexité du problème qui consiste à créer des groupes. Par exemple, le tryptophane peut être considéré comme hydrophobe

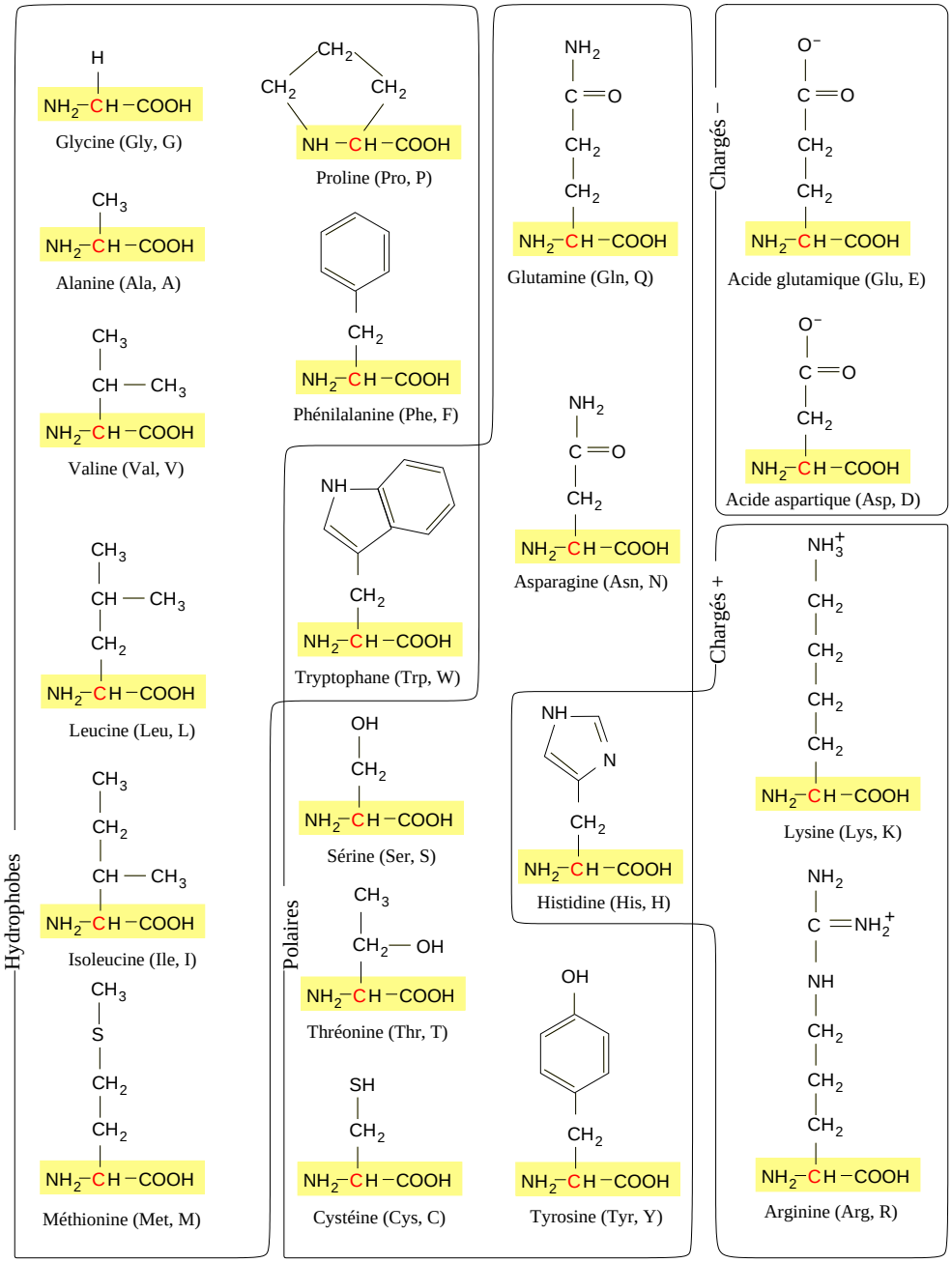

FIG. 1.1 – Les 20 acides aminés standards, groupés d'après les propriétés physico-chimiques des chaînes latérales [27]. Les atomes communs à tous les acides aminés sont représentés sur fond jaune, avec les Cα en rouge. Dans les conditions physiologiques, les fonctions amines et carboxyles libres de la partie jaune sont chargées. Le nom de chaque acide aminé est suivi de son code usuel dans la nomenclature à 3 lettres et à une lettre.

(en raison de son cycle aromatique) ou polaire (en raison de son atome d'azote). L'histidine peut être ou non chargée positivement selon le pH, d'où son inclusion dans deux groupes.

L'hydrophobicité est probablement mieux représentée par l'utilisation d'échelles d'hy-

drophobicité que par la constitution de classes. Plusieurs dizaines d'indices d'hydrophobicité ont ainsi été publiés à ce jour, voir par exemple [45, 22, 99].

D'autres caractéristiques, comme la taille de la chaîne latérale, permettent d'établir des groupes. Taylor a ainsi proposé en 1986 une classification plus fine, avec des recoupements entre classes, représentée sur la figure 1.2 [189]. La cystéine est un cas particulier : au sein des structures 3D, deux cystéines proches peuvent s'apparier pour former une liaison covalente appelée pont disulfure. Les cystéines ainsi appariées ne sont plus polaires. Ces ponts disulfures sont, par ailleurs, particulièrement importants pour le maintien des structures 3D.

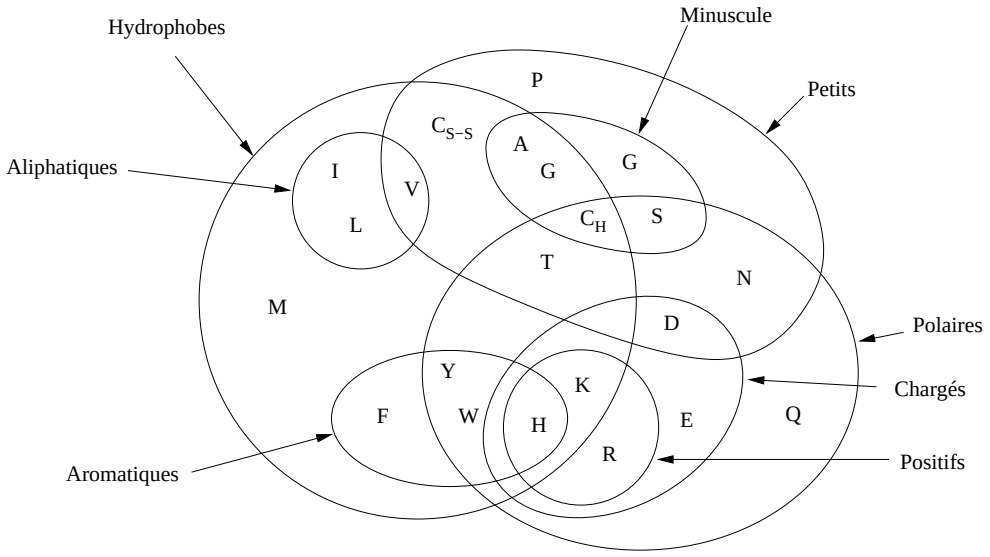

FIG. 1.2 – Diagramme illustrant la classification des acides aminés d'après leurs propriétés physico-chimiques, d'après [189]. La cystéine est présente sous forme non appariée (C_H) et appariée (C_{S-S}).

Liaison peptidique et angles dièdres Φ/Ψ

Une séquence de protéine est formée par un enchaînement d'acides aminés, selon un ordre défini par la séquence d'ADN du gène correspondant. Une protéine de taille moyenne est constituée d'environ 200 acides aminés. La polymérisation de la chaîne protéique se fait par la perte d'une molécule d'eau lors de la condensation d'un groupement carboxyle $COOH$ avec le groupement amine NH_2 du résidu suivant. La liaison ainsi formée, illustrée dans la figure 1.3 est dénommée liaison peptidique et l'acide aminé ainsi incorporé à la

chaîne est dénommé résidu. Les atomes participant à la liaison peptidique forment le squelette de la protéine ou encore la chaîne principale.

La liaison peptidique est rigide en raison de la délocalisation des électrons du groupe carboxyle. Les liaisons C=0 et N-H sont donc maintenues dans le même plan. La grande majorité des liaisons peptidiques dans les protéines sont de type *trans* : les groupes CO et NH pointent dans des directions opposées. De même, les longueurs des liaisons chimiques varient très peu autour de leurs valeurs de référence. Les degrés de liberté pour le repliement de la chaîne protéique correspondent aux rotations autour des liaisons $NH - C\alpha$ et $C\alpha - CO$. Ces deux rotations sont décrites respectivement par les angles dièdres Φ et Ψ, comme illustré sur la figure 1.3.

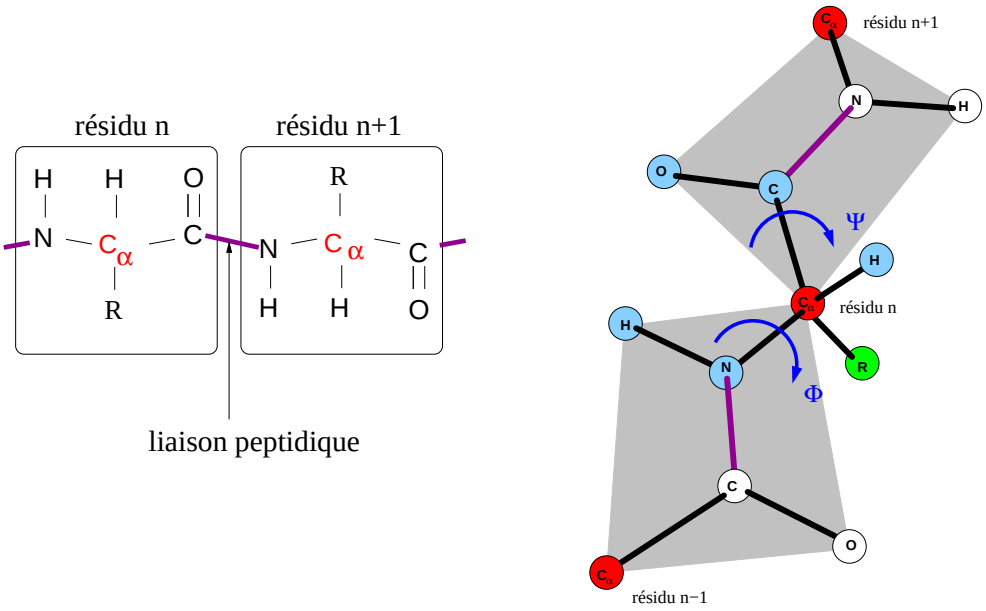

FIG. 1.3 – Liaison peptidique et angles dièdres Φ/Ψ

Diagramme de Ramachadran Les volumes des chaînes latérales limitent considérablement les valeurs effectivement accessibles aux angles Φ/Ψ. Cette limitation du nombre de conformations accessibles a été prédite en 1968 par Ramachandran [150], qui a introduit à l'occasion une représentation par paires des angles encadrant un Cα, sur un diagramme en deux dimensions. Le diagramme de Ramachandran est depuis couramment utilisé pour analyser la structure des protéines. Comme l'illustre la figure 1.4, certaines zones du diagramme sont très fortement défavorables. La proline et la glycine ont des

localisations particulières. La glycine a accès à un plus grand nombre de conformations que les autres résidus, en raison de sa chaîne latérale très réduite. Au contraire, dans le cas de la proline, l'inclusion de l'azote de la chaîne principale dans le cycle de la chaîne latérale introduit une contrainte supplémentaire et limite davantage les valeurs prises par les angles Φ.

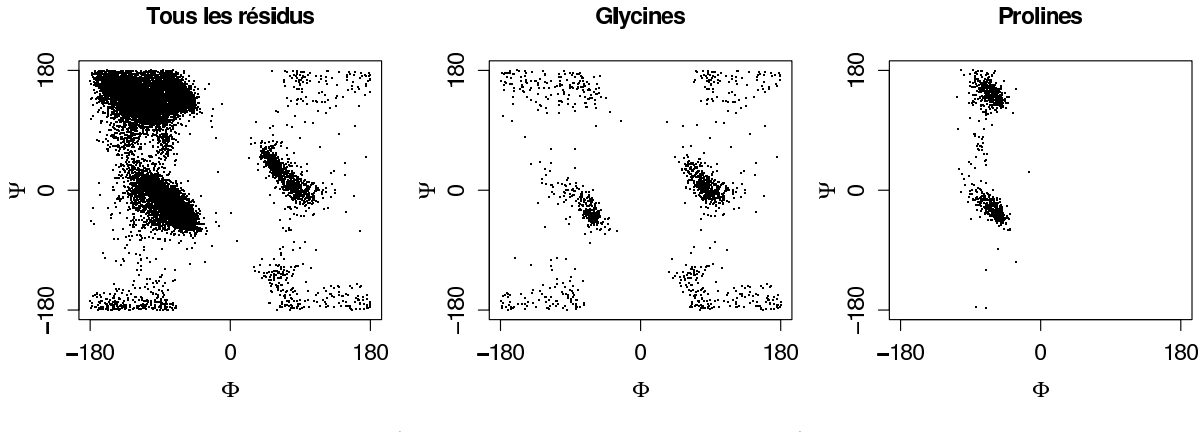

FIG. 1.4 – Diagrammes de Ramachandran. Les paires d'angles dièdres sont extraites d'un ensemble non-redondant de 100 structures.

1.1.2 Structure secondaire

La structure secondaire désigne la conformation adoptée par la chaîne principale au niveau local. L'existence de l'hélice α et du feuillet β, qui sont les deux principales structures secondaires régulières, a été prédite par Pauling et Corey en 1951 [143, 142]. Pauling et Corey recherchaient alors des structures locales régulières permettant de former le maximum de liaisons hydrogènes, liaisons chimiques de faible énergie, au sein du squelette, tout en respectant les longueurs et angles de liaison connus. Ces deux motifs structuraux sont représentés sur la figure 1.5.

L'hélice α est une hélice droite stabilisée par des liaisons hydrogènes entre le groupe carbonyle d'un résidu i et le groupe amide du résidu $i + 4$. L'hélice α a une périodicité de 3.6 résidus. Les chaînes latérales sont projetées à l'extérieur de l'hélice. L'hélice α est la structure secondaire la plus abondante puisqu'elle concerne environ 30% des résidus en moyenne.

Le feuillet β est stabilisé par des liaisons hydrogènes entre des résidus éloignés le long de la séquence, dans des portions de chaîne en conformation étendue, les brins β. Deux catégories de feuillets sont distinguées, selon l'orientation relative des brins : feuillets parallèles et feuillets antiparallèles. Les chaînes latérales pointent alternativement d'un côté et de l'autre du feuillet. La nature non-locale des liaisons hydrogènes dans les feuillets β les différencie fondamentalement des hélices α. Des brins β isolés, existant de manière stable hors des feuillets, ont également été décrits [63]. Environ 20% des résidus des protéines sont impliqués dans des feuillets β.

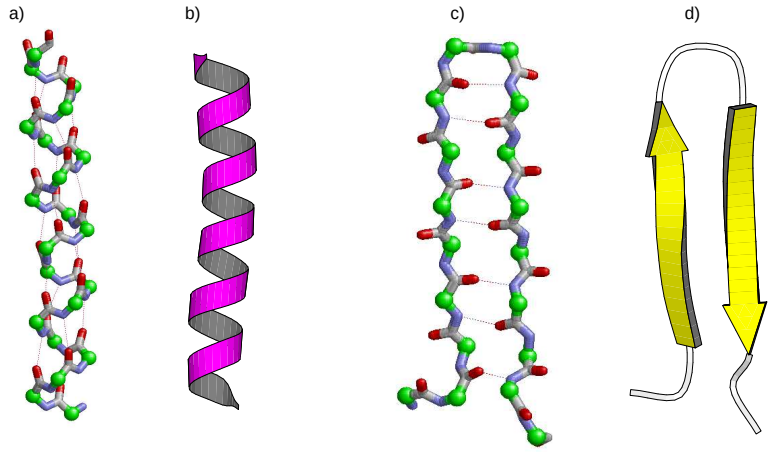

FIG. 1.5 – Hélice α (a, b) et feuillet β parallèle constitué de deux brins (c, d). Les figures a et c représentent la chaîne principale avec les Cα en vert et les liaisons hydrogènes en pointillé. Elles sont réalisées avec le logiciel de visualisation rasmol [167]. Les figures b et d sont des représentations simplifiées en style *cartoon*, générées avec le logiciel molscript [107].

L'apériodique ou boucle (en anglais *coil*), comprend, par défaut, tous les résidus en conformation non-α et non-β. Une description plus fine du coil fait apparaître d'autres motifs réguliers, illustrés dans la figure 1.6, par exemple :

- Le coude β est formé de 4 résidus, stabilisés ou non par une liaison hydrogène [119, 87]. Cette structure courte permet un changement de direction de la chaîne principale et concerne 25 % des résidus [91].
- L'hélice 3-10 est une hélice droite stabilisée par des liaisons hydrogènes de type $(i, i+3)$ [52]. Majoritairement courtes, les hélices 3-10 sont fréquemment rencontrées aux extrémités des hélices α et concernent 3 à 4% des résidus [15].

– L'hélice π est une hélice droite formée par des liaisons hydrogènes de type $(i, i+5)$ [122]. Elle concerne moins de 1% des résidus [67].

Même avec cette classification enrichie, un grand nombre de résidus restent non décrits.

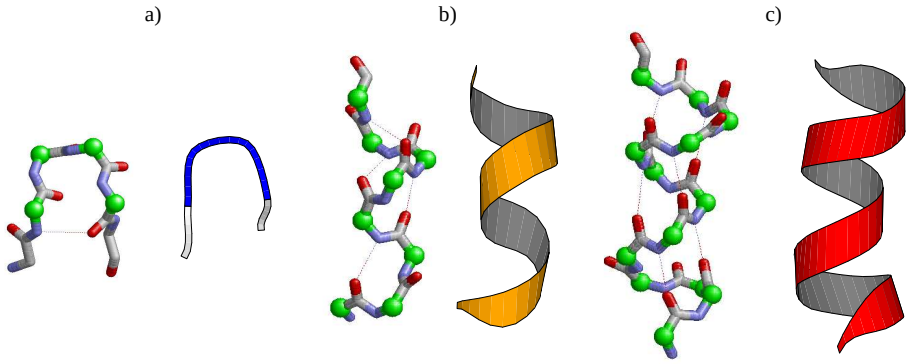

FIG. 1.6 – Coude β (a), hélice 3-10 (b) et hélice π (c).

Les structures secondaires sont aujourd'hui largement utilisées dans un grand nombre d'applications en biologie structurale, car elles permettent une description simple et intuitive des structures 3D. La description des structures 3D en terme de structures secondaires est ainsi mise à profit, par exemple, en comparaison et en classification de structures [76, 133, 137]. Les logiciels de visualisation de structures proposent systématiquement une représentation des structures 3D mettant en évidence les structures secondaires [167, 85] : hélice α, feuillet β et éventuellement coudes.

Irrégularités des structures secondaires

L'observation des structures secondaires périodiques dans les structures disponibles fait apparaître un grand nombre d'irrégularités. Par exemple, la grande majorité des hélices α ne sont pas parfaitement linéaires, mais courbées à des degrés variés, voir coudées [15, 113]. Des π-bulges, provoqués par des liaisons hydrogènes de type $(i, i+5)$ dans les hélices α ont également été décrits [39].

Les feuillets β comportent des altérations dans la régularité de l'appariement des brins, introduites par l'insertion d'un résidu dans l'un des brins : les β-bulges [152]. Les β-bulges sont relativement fréquents dans les feuillets antiparallèles [153] : l'étude de Chan et al [40] dénombre ainsi deux β-bulges par protéine. Ces β-bulges introduisent une discontinuité

dans l'alternance d'orientation des chaînes latérales d'un côté et de l'autre du feuillet. Il a été suggéré que les β-bulges permettent d'accommoder une insertion dans la séquence sans perturber l'architecture globale du feuillet et pourraient également permettre d'éviter l'appariement indésirable avec un brin situé en bordure de feuillet [153]. Ces deux types de bulges sont illustrés dans la figure 1.7.

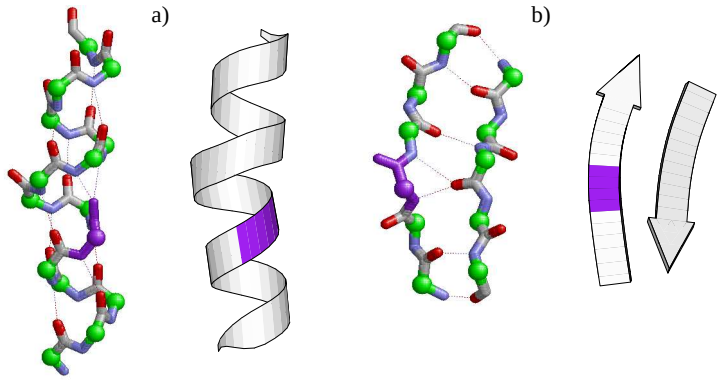

FIG. 1.7 – π-bulge (a) et β-bulge (b). Les résidus responsables des bulges sont en violet. Le π-bulge est extrait de la structure PDB 1c3w (résidus 208-222, chaîne A) et le β-bulge de la structure PDB 1beb (résidus 42-59, chaîne A).

Les motifs de structures secondaires sont donc souvent sensiblement différents des modèles théoriques proposés par Pauling et Corey. Ces irrégularités peuvent, dans certains cas, rendre délicate la détection automatique des structures secondaires. Cet aspect sera développé dans le chapitre 2.

1.1.3 Structure tertiaire

La structure tertiaire d'une protéine est définie par la structure globale tridimensionnelle de la chaîne, y compris la position des atomes des chaînes latérales.

L'ensemble des structures disponibles déterminées expérimentalement est regroupé dans la Protein Data Bank (PDB) [21], `http://www.pdb.org/`. Un fichier PDB contient l'information de structure sous la forme des coordonnées de tous les atomes de la protéine dans l'espace à trois dimensions. Les atomes d'hydrogènes ne sont en général pas décrits dans les structures cristallographiques. Certains atomes de la protéine peuvent ne pas être décrits si les données expérimentales ne permettent pas de déterminer leurs coordonnées.

La PDB comporte à ce jour plus de 30 000 structures déterminées par cristallographie aux rayons X -pour plus de 80% d'entre elles- et par résonance magnétique nucléaire (RMN). De même que les banques de séquences, la PDB est redondante : elle contient de nombreuses structures de protéines dont les séquences sont très similaires, voir identiques (cas de protéines co-cristallisées avec différents inhibiteurs, par exemple).

Relation entre structure 3D et séquence : propriété d'homologie des protéines

Deux protéines sont dites homologues si elles dérivent d'un ancêtre moléculaire commun. Au cours de l'évolution, des mutations s'opèrent sur les séquences d'ADN. Ces mutations sont conservées si les protéines codées sur les gènes conservent leurs fonctions et donc leurs structure tridimensionnelle, en raison de la pression de sélection qui tend à maintenir la fonction. La conséquence de cette pression sélective est que des séquences différentes peuvent adopter la même structure, comme illustré sur la figure 1.8. L'homologie est généralement mise en évidence par la comparaison de séquences, quand l'ancêtre commun est encore assez proche. Si les séquences ont trop divergé, la comparaison de séquences ne permet pas de détecter l'homologie.

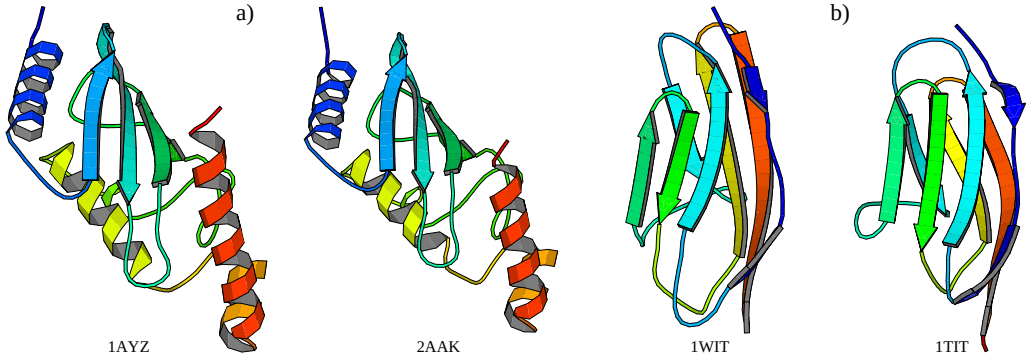

FIG. 1.8 – Exemple de paires de protéines homologues. En a, le cas de deux enzymes qui assurent la même fonction chez la levure et chez le blé (structures PDB 1ayz et 2aak). Les deux séquences sont identiques à 63%. En b, le cas de deux protéines constituant du muscle, chez le nématode et chez l'homme (structures PDB 1wit et 1tit). L'identité de séquences n'est que de 9%.

1.1.4 Structure quaternaire

Plusieurs protéines peuvent s'associer pour former un complexe stabilisé par des liaisons inter-chaînes. De nombreuses protéines sont multimériques : elles sont formées de

plusieurs chaînes, éventuellement identiques.

1.2 Prédiction de la structure des protéines

Les projets de séquençage des génomes génèrent une masse de données considérable. Or, s'il est relativement rapide d'obtenir la séquence d'une protéine, la détermination expérimentale de la structure est difficile, longue et coûteuse. Le nombre de séquences protéiques disponibles dans la banque de données PIR-NREF [203][1]est de l'ordre de 2.4 millions, alors que le nombre de structures disponibles dans la PDB est de l'ordre de 30 000. La structure d'une protéine n'est donc déterminée que pour 1% des séquences connues.

Ainsi, un champ de la bioinformatique consiste à prédire la structure *in silico*. Il faut, tout d'abord, distinguer les méthodes qui visent à prédire localement la structure des protéines, des méthodes visant à fournir un modèle global. Ces deux approches sont liées, puisque la prédiction locale fournit bien souvent les premiers éléments permettant de réaliser une prédiction globale.

La suite de ce chapitre présente les méthodes répondant à ces deux questions.

1.2.1 Prédiction de la structure locale des protéines

Le terme prédiction de structure locale englobe la prédiction de structures secondaires mais aussi d'autres types de prédiction comme la prédiction d'alphabets structuraux ou d'angles Φ/Ψ.

Prédiction de structure secondaire

La prédiction des structures secondaires d'après les séquences constitue un vaste domaine d'étude depuis les années 70. Le nombre impressionnant d'articles publiés sur le sujet interdit de faire une liste exhaustive de toutes les méthodes de prédiction. Seront présentées ici les principales avancées du domaine, ainsi que quelques méthodes choisies.

[1]La banque PIR-NREF est une banque non redondante qui regroupe les séquences de protéines déposées dans les diverses banques de protéines ainsi que celles obtenues par traduction de régions codantes des séquences nucléiques.

Avancées du domaine Les premières méthodes publiées utilisent le fait que, dans les structures tridimensionnelles, certains acides aminés sont retrouvés préférentiellement dans certaines classes de structures secondaires [41, 156, 72]. Les préférences établies sont combinées par des règles de prédiction qui tiennent compte des structures prédites pour les sites voisins. Le taux de bonne prédiction de ces méthode, noté Q_3 et correspondant au pourcentage de résidus correctement prédits en hélice/feuillet/ coil, est de l'ordre de 55 à 58%.

Les méthodes développées dans les années 80 et jusqu'au début des années 90 tiennent compte, dans le processus de prédiction, de l'environnement local des résidus dans la séquence. Pour cela, la prédiction de la conformation d'un résidu est réalisée à partir d'une fenêtre glissante dans la séquence. La taille de cette fenêtre est variable selon les méthodes (3 à 51 résidus [160]).

La croissance des banques de données de séquences a permis d'utiliser l'information des séquences homologues. Les protéines homologues ayant des structures 3D et *a fortiori* des structures secondaires similaires, l'idée est de faire la prédiction en utilisant les séquences homologues de la protéine à prédire. Cette information est le plus souvent prise en compte par l'utilisation d'un profil. Un profil est un tableau contenant les fréquences des acides aminés présents dans les colonnes d'un alignement multiple de séquences. La prise en compte des séquences homologues permet d'atteindre des Q_3 supérieurs à 75%.

Algorithmes Une grande variété d'algorithmes a été appliquée au problème de la prédiction de structure secondaire.

La théorie de l'information est à la base des méthodes GOR (Garnier, Osguthorpe et Robson) [72, 74, 71] : elle permet de formuler l'influence de la séquence locale sur la conformation des résidus de manière rigoureuse.

Un grand nombre de méthodes utilisent les réseaux de neurones[2] depuis les travaux précurseurs de Qian et Sejnowski [148]. La méthode PHD de Rost et Sander [162] utilise une fenêtre de 13 résidus dans la séquence et une prédiction en trois étapes : un pre-

[2]Un réseau de neurone est un système combinant des éléments simples, les neurones, qui fournissent une *réponse* aux signaux d'entrée émis par leurs voisins. Ce modèle est très schématiquement inspiré du fonctionnement des neurones biologiques. Pour la prédiction de structure secondaire, un réseau de neurones est entraîné dans le but d'*apprendre* à classer les séquences en hélice/brin/coil, par minimisation de l'erreur quadratique commise.

mier réseau de neurones (*sequence-to-structure*) prédit la structure secondaire du résidu central, un deuxième réseau (*structure-to-structure*) reçoit en entrée une fenêtre de 17 positions dans la prédiction fournie par le premier réseau et renvoie la structure secondaire du résidu central, enfin la moyenne est réalisée sur plusieurs réseaux de neurones (*jury decision*). L'information d'entrée est constituée par les profils issus des alignements multiples de la base HSSP [166], ainsi que des poids relatifs à la conservation dans les colonnes de l'alignement. La méthode PSIPRED de D. Jones [90] utilise une fenêtre de taille 15 et effectue également une correction des résultats du réseau par un deuxième réseau *structure-to-structure*. L'information des séquences homologues est prise en compte sous la forme de profils générés par le programme d'alignement de séquences PSI-BLAST [4]. La méthode SSPRO de Baldi et al utilise des réseaux de neurones particuliers appeler réseaux bidirectionnels récurrents [12, 146]. Riis et Krogh spécifient des réseaux pour chaque classe structurale à prédire [154].

Les méthodes de type *plus proches voisins* utilisent les structures de protéines disponibles pour inférer la structure secondaire par comparaison de fragments [117, 164, 65, 89, 104]. La méthode SIMPA [117, 116] utilise les scores d'alignements locaux pour effectuer la prédiction. Dans la méthode SOPMA [73], les paramètres de prédiction sont optimisés pour la protéine à prédire, en utilisant des structures connues similaires. Ces deux méthodes appliquent un algorithme de régularisation des prédictions [211].

Plus récemment, des méthodes basées sur les SVM (Support Vector Machine, en français, Séparateurs à Vaste Marge [3]) ont été proposées [84, 103, 198, 83].

Quelques auteurs ont proposé des méthodes basées sur les modèles de chaînes de Markov cachées : Asai et al en 1993 [6], Stultz et al en 1994 [186, 200], Jones et al en 1996 [192], [121] et Crooks et Brenner en 2004 [46]. Ces méthodes seront détaillées dans le chapitre 3.

Parmi les autres approches, on trouve, entre autre, les statistiques multivariées [94], la programmation logique inductive [131, 47], ou encore la mécanique moléculaire [100].

[3] Un SVM est une méthode permettant d'établir un hyperplan séparant au mieux des points (exemples positifs *vs* exemples négatifs). Le principe des SVM est de projeter les points dans un espace de dimension supérieure dans lequel il existe un séparateur linéaire qui permet de classer les points. En prédiction de structure secondaire, les méthodes mettant en œuvre des SVM utilisent souvent plusieurs classifieurs binaires (α/non-α, β/non-β, coil/non-coil.)

Méthodes consensus Dès 1988, des approches consistant à rechercher un consensus entre plusieurs algorithmes de prédiction ont été proposées [22]. Le but est de corriger les biais intrinsèques à chaque méthode, en effectuant la prédiction par plusieurs algorithmes puis en combinant les prédictions. Ainsi, en combinant 3 méthodes, Biou et collaborateurs ont montré qu'il est possible d'améliorer le Q_3 de 2.5 à 6.5 points par rapport aux prédictions des méthodes individuelles. PROF_King de Ouali et King [139] est une méthode consensus qui utilise des classifieurs en cascade -dont les méthodes GOR- en plus de réseaux de neurones originaux développés par les auteurs. Le Q_3 est ainsi amélioré de plus de 10 points, atteignant 76.7%. La méthode HYPROSP [204] combine PSIPRED et une banque de petits fragments de structure connue. Ceci permet d'améliorer la prédiction de PSIPRED pour les protéines ayant une forte similarité locale avec la banque de fragments. Les travaux de Guermeur et al [79] utilisent les SVM multiclasses pour combiner GOR, SOPMA et SIMPA, ou les réseaux de neurones de SSPRO.

Evaluation des méthodes de prédiction de structures secondaires Comparer objectivement entre elles les différentes méthodes n'est pas une tâche triviale, ceci pour plusieurs raisons.

- **La référence utilisée**, autrement dit l'assignation de structures secondaires considérée comme standard de vérité, doit être identique pour toutes les méthodes. L'assignation utilisée est généralement celle fournie par le programme DSSP [91]. DSSP assigne initialement 8 conformations locales : hélice α, brin β, hélice 3-10, hélice π, coude, bend (région courbée), β-bridge isolé et coil. Les méthodes de prédiction fournissant habituellement une prédiction à trois modalités, il est nécessaire de réduire ces 8 états en 3. La réduction adoptée influence les taux de performances [159].

- **Les scores utilisés** pour quantifier la prédiction doivent refléter au mieux les apports et lacunes de chaque méthode. En général les auteurs rapportent le score Q_3, qui est le pourcentage de résidu prédits dans la bonne conformation. D'autres indices sont plus appropriés, comme le score SOV [208] qui tient compte du recouvrement en terme de segments de structures secondaires.

- **La redondance des jeux de données** utilisés pour l'entraînement et le test des méthodes peut biaiser l'évaluation des méthodes. D'une part, si le jeu d'apprentissage contient des séquences homologues, les paramètres risquent d'être biaisés vers les familles

de séquences les plus représentées. D'autre part, si le jeu de test présente des homologies avec le jeu d'apprentissage, les performances risquent d'être sur-évaluées.

- **Les jeux de test** peuvent être intrinsèquement plus ou moins difficiles à prédire. Par exemple, la prédiction des hélices α étant plus facile, les méthodes testées sur des protéines tout α affichent de meilleurs résultats. D'une manière générale, il est délicat de comparer des méthodes évaluées sur différents jeux de données. Avec une méthode donnée, en prenant les précautions nécessaires sur la redondance, on peut obtenir des performances différentes sur deux jeux de tests non corrélés. Ainsi, B. Rost rapporte avoir obtenu des performances significativement différentes en testant sa méthode PROF sur deux jeux de données distincts, non corrélés entre eux et non corrélés au jeu d'apprentissage [159].

- **La date à laquelle a été effectué le test** modifie les résultats des méthodes utilisant les profils de séquences. En effet, le contenu des banques de séquences étant en perpétuelle augmentation, les profils s'enrichissent et les méthodes utilisant les profils voient leur performances s'améliorer par ce seul effet. Cette précaution semble parfois négligée au moment de comparer les performances d'une nouvelle méthode avec celles publiées pour les méthodes standards [103].

A cet égard, une initiative intéressante est constituée par le serveur web EVA [106]. Les méthodes inscrites à EVA sont testées en continu sur les nouvelles structures déposées dans la PDB. Seules les protéines qui n'ont pas d'homologie détectable avec les protéines déjà déposées dans la PDB sont utilisées pour l'évaluation, ce qui permet d'éviter les biais de sur-apprentissage. Les structures secondaires de référence et les indices utilisés sont les mêmes pour toutes les méthodes.

En revanche, le nombre de séquences analysées dépend de la date à laquelle les méthodes ont été incluses dans la comparaison, ce qui ne permet pas de comparer les performances de toutes les méthodes entre elles. Peu de détails sont donnés sur les précautions prises pour contrôler la composition en structure secondaire des séquences testées. D'autre part, cette comparaison assure artificiellement une meilleure visibilité aux méthodes participantes.

Compte-tenu de ces réserves, les résultats d'EVA montrent que, sur des jeux de données de plus de 100 structures, les meilleures méthodes, en général basées sur les réseaux de neurones, atteignent des Q_3 de 78%. Les résultats obtenus sur des jeux de données communs de plus de 100 protéines sont récapitulés dans le tableau 1.1.

Jeu	1	2	3
Nb struct	161	165	182
Classé 1	PSIpred(77.7/75.9) SAM-T99sec(77.5/74.9))	PROFsec(76.6/75.3) PSIpred(77.6/75.8) SAMT99sec(77.4/74.7)	PSIpred(78.0/75.8)
Classé 2	PROFsec(76.4/75.3)	PHD(74.4/70.5) PHDpsi(74.6/70.6)	PROFsec(76.8/75.0)
Classé 3	PHD(74.4/70.7) PHDpsi(74.6/70.8)		PHD(75.0/70.7) PHDpsi(75.1/70.8)

TAB. 1.1 – Classement des meilleures méthodes évaluées sur le serveur EVA, d'après `http://salilab.org/~eva/sec/common.html`. Nb struct : nombre de structures dans le jeu de données correspondant (non détaillé). Les deux nombres rapportés entre parenthèse sont respectivement le score Q_3 et le score SOV [208] qui tient compte du recouvrement des segments de structures secondaires.

PHDpsi [147] est une évolution de PHD utilisant les profils fournis par PSI-BLAST.

SAMT99sec est un dérivé de SAM-T99, une méthode de reconnaissance de repliement utilisant les modèles de chaînes de Markov cachées [97]. Cette méthode n'a pas fait l'objet d'une publication en tant que telle. Il semble que la prédiction de structure secondaire soit en fait réalisée par un réseau de neurones [4] [95].

Des méthodes plus récentes, comme SSPRO, PORTER [145], ou SABLE [1], toutes basées sur les réseaux de neurones semblent améliorer encore ces performances, mais elles sont testées sur trop peu de séquences à l'heure actuelle.

Prédiction d'alphabets structuraux

La description et la prédiction de structures secondaires ne fournit aucune indication structurale sur l'apériodique. L'utilisation d'alphabets structuraux apporte une description complète des structures. Un alphabet structural peut être vu comme un jeu de Lego, dont les briques permettent de reconstruire les structures 3D existantes.

Les alphabets structuraux répondent à un double objectif :

– fournir une description précise des structures, dans un but d'analyse ou de reconstruction,

[4] `http://www.cse.ucsc.edu/research/compbio/SAM_T02/sam-t02-faq.html`

– être utilisés pour la prédiction de structure locale.

La classification systématique de petits fragments extraits des structures 3D des protéines a ainsi montré qu'il existe des frontières entre classes de fragments et a mis en évidence la corrélation entre les prototypes structuraux et les séquences en acides aminés sous-jacentes[158].

Il a été montré qu'un nombre limité de prototypes structuraux permet d'approximer correctement les structures 3D existantes, par exemple, 100 prototypes de 6 résidus dans l'étude de Unger et al [194]. La précision de la reconstruction dépend de la longueur et du nombre de prototypes. Une librairie de 100 fragments de 5 résidus permet d'approximer les structures 3D avec une erreur moyenne de 0.9 Å sur la position des Cα ; cette erreur est de 1.8 Å avec 20 fragments de 5 résidus [101]. Plusieurs équipes ont ainsi mis au point des alphabets structuraux dans le but d'optimiser la reconstruction [141, 129].

Dans une optique de prédiction, des alphabets structuraux de taille plus réduite ont ensuite été établis : 27 blocs de 4 résidus pour Camproux et al [38] dans une classification utilisant les modèles de chaînes Markov cachées, 16 blocs de 5 résidus pour de Brevern et al [49], 28 blocs de 7 résidus pour Hunter et Subramanian [86]. Cette variété rend la comparaison des méthodes de prédiction difficile.

Prédiction des angles Φ/Ψ

Les angles Φ/Ψ permettent de décrire le cheminement de la chaîne principale. Contrairement à la description en terme de structure secondaire, ils apportent une information sur la conformation des résidus en coil.

Prédiction des angles Φ/Ψ par la théorie de l'information, Gibrat et al, 1991 [75] La théorie de l'information est utilisée pour prédire les 4 zones d'angles Φ/Ψ du diagramme de Ramachandran illustrées sur la figure 1.9.

Après l'encodage en zones d'angles, 61 structures sont utilisées pour dériver les paramètres de l'information nécessaires au calcul des informations I pour chacune des 4 zones d'angles :

$$I(S_j; R_{j-8}...R_{j-8}) = ln\frac{P(S_j/R_{j-8}...R_{j-8})}{P(S_j)}$$

$I(S_j; R_{j-8}...R_{j-8})$ représente l'information apportée par la séquence locale de 17 résidus $R_{j-8}...R_{j-8}$ sur la conformation S_j du résidu central R_j. $P(S_j)$ est la probabilité *a priori*

de la conformation S_j, estimée par sa fréquence dans la base de données. De même, $P(S_j/R_{j-8}...R_{j-8})$ est la probabilité de la conformation S_j au centre de la séquence locale $R_{j-8}...R_{j-8}$. L'estimation de cette dernière probabilité nécessite des approximations car la quantité de données disponible ne contenait pas assez d'exemples de toutes les séquences possibles de 17 résidus. La prédiction est fournie par la zone d'angle dont l'information mutuelle est la plus forte.

Les paramètres calculés permettent de prédire la bonne zone d'angle pour 62% des résidus. Les zones g et x sont les moins bien prédites.

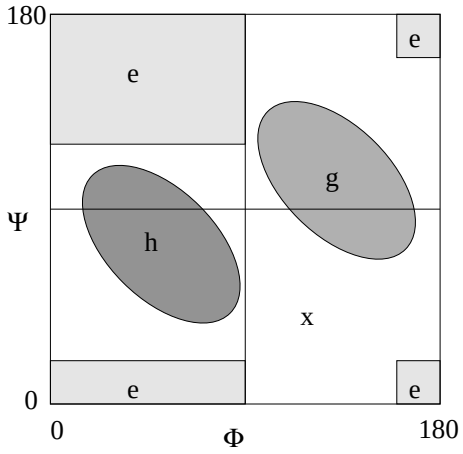

FIG. 1.9 – Les 4 zones d'angles h, g, e et x définies par Gibrat et al [75]. Pour la glycine, la zone e est étendue. La zone x concerne très peu de résidus.

Rooman et al, 1991[157] Rooman et al ont utilisé un encodage en 7 zones d'angles (figure 1.10) pour prédire la structure 3D de petites protéines. La prédiction utilise la théorie de l'information et les champs moyens.

La précision de prédiction en terme de zones d'angles n'est pas rapportée.

Prédiction d'après la banque de fragments I-SITES et un modèle de chaînes de Markov cachées, 2000 [36] La librairie de fragments structuraux I-sites a été construite par classification de tous les fragments chevauchants de 3 à 15 résidus d'une banque non redondante de 471 structures, d'après la comparaison locale de leurs profils de séquence [34]. Au sein de chaque classe, le fragment ayant la structure la plus représentative est élu structure paradigme. Les 82 classes ainsi obtenues sont raffinées sur des bases structurales, en retirant les fragments trop éloignés de la structure paradigme. La

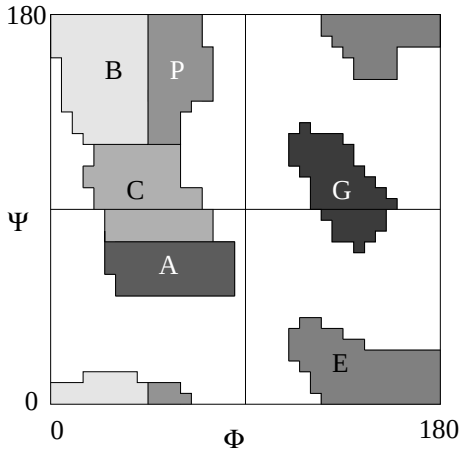

FIG. 1.10 – Zones d'angles utilisées par Rooman et al [157]. Une septième zone, non montrée, correspond aux résidus en conformation *cis*.

stabilité structurale des classes est ainsi renforcée. Le but de cette classification est d'obtenir une librairie de fragments présentant une forte corrélation séquence/structure. Cette librairie a ensuite été étendue en *masquant* les fragments fortement corrélés aux I-sites existants pour identifier de nouveaux motifs exhibant des corrélations séquence/structure plus faibles [36].

La librairie enrichie de 262 I-sites est utilisée pour construire des modèles de chaînes de Markov cachées qui tiennent compte des connexions préférentielles entre motifs. La construction et l'utilisation des ces modèles seront précisées dans le chapitre 3. L'un des modèles proposés est utilisé pour la prédiction de 11 zones d'angles illustrées dans la figure 1.11.

La performance de prédiction est de 56% pour les 11 zones. La zone c n'est jamais prédite. Les zones peu peuplées b, d, e ainsi que L et G sont sous-prédites. Quand ces 11 zones sont regroupées en 4 classes (H+G, B+E+b+d+e, L+I, x), le taux de bonne prédiction est de 74 % et monte à 75% en groupant les zones L+I et x.

Prédiction consensus utilisant la banque de fragments LSBSP1, Yang et Wang, 2003 [206] Une librairie de fragments appelée LSBSP1 est utilisée pour fournir une prédiction des angles Φ/Ψ. La banque LSBSP1 est constituée par classification structurale des fragments de 9 résidus extraits d'une banque de structures non redondante. La similarité de structure a été évaluée par un encodage en 11 zones d'angles Φ/Ψ dans le diagramme de Ramachandran d'après Oliva et al [136]. Les segments d'encodage iden-

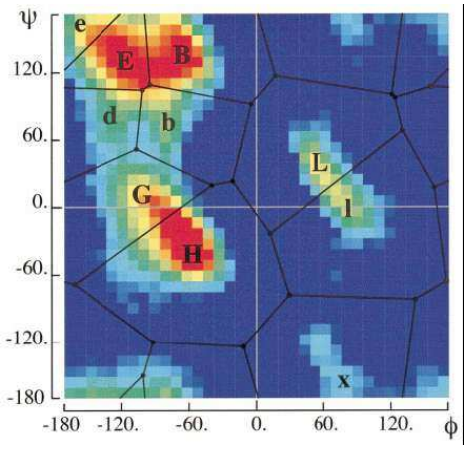

FIG. 1.11 – Zones d'angles utilisées par Bystroff et al. Une onzième zone (c), non montrée, correspond aux résidus en conformation *cis*. Figure extraite de [36].

tique ayant une similarité de séquence suffisante avec le fragment initial sont inclus dans le groupe. La comparaison de séquences utilise des matrices de substitution spécifiques des zones d'angles. Ces groupes sont ensuite convertis en profils de séquences. Ces profils sont utilisés pour raffiner les groupes en excluant les segments de similarité de séquence insuffisante avec l'ensemble du groupe, puis pour récupérer les segments d'encodage identiques qui n'avaient pas été récupérés dans la première étape. Ceci permet l'obtention de la banque finale LSBSP1. Le but est d'opérer une classification guidée à la fois par la séquence et la structure. L'étape de raffinage est destinée à renforcer la similarité de séquence au sein de chaque groupe, car des segments d'encodage structural identique peuvent appartenir à des familles de séquences différentes.

La procédure de prédiction est la suivante : (i) La structure secondaire de la séquence candidate est prédite avec PSIPRED. La séquence est divisée en segments chevauchants de 9 résidus. La prédiction est ensuite réalisée pour chaque segment. (ii) Les candidats structuraux de LSBSP1 sont sélectionnés en comparant la séquence du segment cible avec les profils de LSBSP1, ainsi que la prédiction de PSIPRED avec la structure secondaires des fragments de LSBSP1. (iii) Un score consensus est calculé à partir des encodages pour chacun des candidats structuraux, afin de sélectionner le candidat ayant l'encodage le plus représenté dans l'ensemble des candidats. (iiii) La séquence cible est alors reconstituée en rassemblant les prédictions locales des segments. Une prédiction consensus est réalisée pour chaque résidu de la séquence cible, d'après les prédictions réalisées pour les segments

chevauchants le long de la séquence cible.

Les performances prédictives sont évaluées sur la prédiction des 4 grandes zones d'angles figurant sur la figure 1.12. En moyenne, la prédiction est possible pour 94.6% des résidus et elle est correcte pour 79.0% des résidus. Dans l'hypothèse où les 5.4% de résidus non prédits recevraient une prédiction incorrecte, le taux de bonne prédiction serait alors de 74.7%. Les zones E' et G' sont les classes les moins bien prédites. Cette méthode repose sur la séquence cible seule, l'information de profil est prise en compte indirectement par PSIPRED.

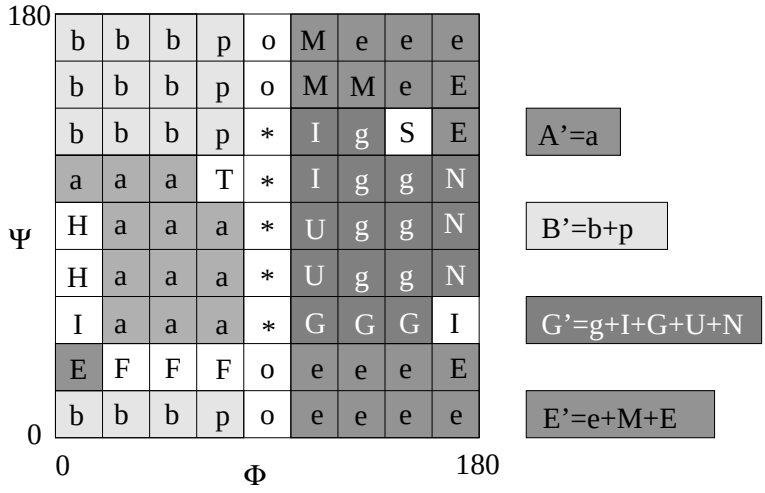

FIG. 1.12 – Les 11 zones d'angles définies par Oliva et al [136] utilisées pour l'encodage des structures (gauche) et réduction en 4 zones utilisées pour la prédiction (droite). D'après [206].

Prédiction utilisant la banque de fragments LSBSP1 et les réseaux de neurones, Yang et al, 2004 [111] La prédiction des angles Φ/Ψ à l'aide de LSBSP1 a été améliorée en utilisant un réseau de neurones. La procédure de prédiction est la même que ci-dessus. Au lieu de réaliser une prédiction consensus (étape iiii), toutes les prédictions locales sont conservées et converties en profil de prédiction. La séquence d'un segment de 9 résidus à prédire, ainsi que son profil de prédiction, est fournie en entrée d'un réseau de neurones qui a été optimisé pour retrouver la structure réelle du résidu central, en terme des 4 zones d'angles : A', B', G' et E' de la figure 1.12. En regroupant les zones G' et E', le taux de bonne prédiction dans ce cas est de 78.7% et de 63.5% si l'on isole les résidus en coil.

Les auteurs présentent également une méthode de prédiction basée sur un SVM entraîné sur les profils fournis par PSI-BLAST et la prédiction de structure secondaire de PSIPRED. Le taux de bonne prédiction est de 77.3% sur les 4 zones d'angles et de 78.7% en regroupant les zones G' et E'. PSIPRED est ici utilisé sans validation croisée.

1.2.2 Prédiction de la structure globale des protéines

Les méthodes de prédiction de structure locale sont une première étape vers la prédiction des structures globales des protéines. Les stratégies employées ne sont pas les mêmes suivant les informations disponibles sur la séquence à prédire.

Modélisation par homologie

La modélisation par homologie repose sur la relation d'homologie entre deux protéines (voir section 1.1.3). L'homologie entre deux protéines est mise en évidence par l'alignement de séquences avec des outils comme PSI-BLAST [4]. Si deux protéines ont une identité de séquences suffisante (environ 25 % de résidus indentiques ou une e-value de PSI-BLAST significative) on peut considérer qu'elles sont homologues.

La modélisation par homologie s'applique donc dans le cas où une structure de protéine homologue de la séquence cible à modéliser est disponible dans la PDB. Cette structure connue, appelée support, est utilisée pour dériver un modèle 3D de la séquence cible. Le programme de modélisation par homologie le plus utilisé est MODELLER [165] (voir aussi le site du Pôle BioInformatique Lyonnais `http://geno3d-pbil.ibcp.fr/`). A partir de l'alignement entre la séquence cible et la séquence support, MODELLER construit un modèle atomique par satisfaction de contraintes spatiales dérivées de l'alignement. La forme de ces contraintes dérive d'une analyse statistique des paires de structures homologues.

L'étape limitante de la modélisation consiste à aligner correctement les deux séquences [195]. La modélisation des boucles variables reste difficile. Dans ce champ, des serveurs automatisés fournissent parfois des modèles aussi satisfaisants que des experts. Il est possible de modéliser les structures par homologie pour 16 à 30% des protéines d'un génome [26].

Reconnaissance de repliements

La reconnaissance de repliement consiste à identifier dans une collection de structures disponibles, un repliement adapté à la séquence à modéliser. La différence avec la modélisation par homologie est que la relation d'homologie n'est pas détectable par les outils d'alignement de séquence, ce qui est le cas quand l'ancêtre commun est trop éloigné. Un autre cas peut se produire : l'évolution convergente de deux protéines vers la même structure, en l'absence d'ancêtre commun.

Des méthodes basées sur les modèles de chaînes de Markov cachées ont été proposées pour détecter l'homologie distante, comme les programmes SAMT de Karplus et al [98, 97] et HMMER de Eddy et al [57]. L'emploi des modèles de chaînes Markov cachées dans ce cadre sera précisé dans le chapitre 3. Une autre approche, comme le logiciel FROST de Marin et al [126] consiste à *enfiler* la séquence cible sur toutes les structures d'une banque et à évaluer la compatibilité entre la séquence et chacune des structures. Cette banque de structures doit être représentative des protéines disponibles. La compatibilité est évaluée au moyen de matrices de scores adaptées au contexte structural, portant sur les résidus et les paires de résidus en contact.

Au final, après le choix d'un support structural et l'alignement de la séquence cible sur le support, le modèle 3D est obtenu à l'aide des outils de modélisation par homologie. La difficulté supplémentaire est de mettre en évidence la compatibilité de la séquence avec une structure existante.

Prédiction de structure *de novo*

Les deux premières approches nécessitent de trouver dans la PDB des structures sur lesquelles il est possible d'*adapter* la séquence à prédire. Les méthodes de prédiction *ab initio* ont pour but de proposer des structures 3D dans les cas où la modélisation par homologie et la reconnaissance de repliement ne fournissent pas de réponse, soit parce que la séquence à prédire adopte une structure qui n'a jamais été observée, soit parce que la structure compatible n'est pas détectée.

***ab initio* ou *de novo* ?** La modélisation *ab initio* consiste à rechercher la structure de la séquence cible en partant uniquement de la séquence. Les approches *de novo* utilisent les structures de la PDB pour extraire des fragments, qui sont ensuite assemblés pour

former un modèle global. Les méthodes *de novo* ne sont donc pas des méthodes *ab initio* au sens strict.

Il est admis que la plupart des structures natives correspondent aux conformations du minimum d'énergie libre. La prédiction *ab initio* consiste à construire un modèle physique simplifié de la chaîne protéique et à effectuer recherche exhaustive de l'espace conformationnel pour obtenir la structure de moindre énergie. Les approches *ab initio* reposent donc sur :

1. une représentation simplifiée de la chaîne,
2. une méthode d'exploration de l'espace conformationnel,
3. une fonction d'énergie à minimiser.

Représentation simplifiée des protéines La protéine peut être représenté uniquement par ses Cα, ou par les Cα et un centroïde représentant chaque chaîne latérale. La représentation HP consiste à distinguer 2 types de résidus : hydrophobes (H) et polaires (P) [51]. Le modèle simplifié est éventuellement enrichi après les premiers stades de la recherche conformationnelle.

Exploration de l'espace conformationnel La recherche conformationnelle est souvent réduite en utilisant un modèle de grille (modèles *lattice*) : les pseudo-atomes du modèle simplifié ne peuvent alors se trouver que sur les noeuds d'un maillage de l'espace 3D [205]. Toutes les méthodes de novo ne reposent pas sur les modèles de *lattice*. Les méthodes utilisées pour explorer l'espace conformationnel sont notamment : la dynamique moléculaire, l'échantillonage par méthode de Monte-Carlo, le recuit simulé, les algorithmes génétiques [138]. Le solvant (l'eau autour de la protéine) peut être pris en compte implicitement ou même négligé [138]. Dans les approches hiérarchiques, certaines parties de la protéine sont *gelées* si leur repliement est jugé satisfaisant [184]. Les profils de séquences et la prédiction de structure secondaire peuvent être utilisées pour introduire des contraintes sur les modèles et restreindre ainsi la recherche conformationnelle [183].

Fonction d'énergie Les fonctions d'énergie dépendent de la représentation utilisée. La représentation HP ne considère que trois types d'énergie : HH, HP et PP ; les contacts HH sont favorisés. L'énergie des modèles *tout-atome* peut être évalué par les champs de

force utilisés en dynamique moléculaire, comme le champs de force CHARMM [32]. Les modèles simplifiés utilisent des potentiels physiques simplifiés et des potentiels statistiques implicites [138].

Malgré les progrès accomplis [80], la prédiction *ab initio* reste très difficile. L'espace conformationnel à explorer est immense. et la fonction d'énergie est particulièrement complexe à formuler.

La stratégie *de novo*, plus récente, est une manière de *découper* le problème. Il s'agit de prédire la structure de petits fragments de la séquence cible, puis de les assembler pour former un modèle global. Il est là aussi nécessaire de se munir d'une représentation de la chaîne, d'une méthode d'exploration de l'espace et d'une fonction d'énergie. L'étape de prédiction de fragments limite considérablement l'espace des conformations possibles. Depuis quelques années, les méthodes *de novo* permettent d'obtenir des modèles de qualité satisfaisante qui respectent la topologie globale des structures cibles [3].

Affinement du modèle

La prédiction de structure globale nécessite souvent une étape d'affinement du modèle. Le positionnement des chaînes latérales peut ainsi être optimisé en utilisant une librairie de rotamères (le terme rotamère désigne la conformation des chaînes latérales) par le programme SCWRL [54]. La qualité du modèle final peut être évaluée en utilisant des potentiels empiriques qui comparent les caractéristiques du modèle, en terme d'environnements locaux de résidus et paires de résidus, aux caractéristiques des protéines de la PDB. Le modèle est ainsi évalué de manière globale, mais ces méthodes permettent aussi de mettre en évidence des parties dont les caractéristiques sont inhabituelles par rapport aux structures connues. Les programmes PROSAII [181] et Verify 3D [123] sont ainsi dédiés à l'évaluation des modèles.

Différents modèles, différentes applications

Les modèles prédits par les trois stratégies présentées ci-dessus sont obtenus avec des informations plus ou moins riches. L'on ne peut donc pas espérer en obtenir la même information [11]. Ainsi, les structures déterminées expérimentalement permettront l'étude des mécanismes catalytiques, qui requièrent une précision atomique. Les modèles prédits par homologie ou par reconnaissance de repliement permettront les études d'interaction

entre protéines, la détection d'épitopes et la mutagénèse dirigée. Avec les prédictions *de novo*, il sera possible d'identifier des groupes de résidus conservés à la surface des protéines et de rechercher des sites fonctionnels.

La compétition CASP et les différentes techniques de prédiction Les compétitions CASP (Critical Assessment of Techniques for Protein Structure Prediction) se tiennent tous les deux ans et permettent une évaluation en aveugle des méthodes de prédiction de structure. Des séquences dont la structure 3D est en cours de résolution sont mises à disposition des groupes de prédiction.

Les séquences à prédire sont divisées en groupe selon la difficulté de la tâche :

- Cibles pour la modélisation par homologie,
- Cibles pour la reconnaissance de repliement,
- Cibles pour la modélisation *de novo*.

Les structures correspondantes ne sont rendues publiques qu'à la fin de la compétition. Ces compétitions connaissent un succès grandissant : le nombre de groupes participant est passé de 35 pour la première édition en 1994 à 201 lors de CASP6 en 2004. Depuis CASP3, la distinction est faite entre les groupes d'experts et les serveurs de prédiction, entièrement automatisée. 65 serveurs ont concouru à CASP6. L'intérêt est de comparer les différentes approches sur les mêmes protéines cibles, sans biais possible, et avec les mêmes critères d'évaluation, ce qui permet à la communauté des prédicteurs d'évaluer les apports méthodologiques des différentes techniques. Le revers de ces évaluation que l'interprétation des résultats est souvent difficile, en raison du faible nombre de cibles. Lors de CASP5 [3], les méthodes les plus performantes sur les cibles *de novo* :

- utilisaient la prédiction des structures secondaires, la plupart du temps par PSI-PRED,
- utilisaient les profils de séquences,
- nécessitaient souvent une intervention humaine pour inspecter les modèles générés.

Jusqu'à sa cinquième édition, en 2002, les méthodes de prédiction de structure secondaire étaient évaluées dans CASP. Depuis, elles sont intégrées dans CAFASP [66] qui est dédié à l'évaluation des serveurs automatiques.

Les travaux de Baker et collaborateurs [179, 34, 178, 180, 24, 30].

Depuis CASP4, le groupe de David Baker s'illustre par de très bons résultats avec la méthode *de novo* Rosetta. La méthode Rosetta repose sur une vision du mécanisme de repliement des protéines du *local vers le global* : la structure locale des protéines est restreinte à un nombre limité de conformations. Des interactions non-locales stabilisent la structure globale de moindre énergie, compatible avec les conformations locales.

L'hypothèse de base de la méthode Rosetta est que la distribution des conformations d'un fragment peut être approchée par la distribution des structures observées pour des fragments de séquence similaire dans la PDB.

Cette méthode, assez simple dans son principe, a fait l'objet d'un remarquable travail de développement à chacune de ses étapes. La prédiction de structure se fait en trois étapes :

1. Prédire la structure de fragments chevauchants de la séquence cible à l'aide de la PDB.
2. Assembler les fragments avec une procédure de Monte-Carlo par tirage dans l'espace des fragments candidats.
3. L'assemblage étant réalisé un grand nombre de fois, choisir les meilleurs modèles.

La méthode est expliquée brièvement ci-dessous, et plus longuement en annexe.

Prédiction de fragments La séquence cible est découpée en segments chevauchants de 3 et 9 résidus. Les candidats structuraux sont identifiés par comparaison de séquences avec les structures de la PDB. Cette étape fournit 25 candidats structuraux par fragment chevauchant de la séquence cible.

Assemblage L'assemblage se fait par une méthode de Monte-Carlo. La protéine est modélisée par un modèle simplifié composé des atomes lourds de la chaîne principale et des carbones β des chaînes latérales. La structure initiale est peu à peu modifiée en substituant les angles dièdres par ceux du fragment tiré dans les candidats possibles. La modification est acceptée si elle n'introduit pas de conflit stérique et ne pénalise pas trop la fonction d'énergie à optimiser. La pénalité acceptable dépend de la *température de simulation*, qui décroît au cours de l'optimisation. De cette façon, de plus grandes

violations sont autorisées au début de l'optimisation. La fonction à optimiser est dérivée d'une fonction de score basée sur la log-vraisemblance :

$$f = -logP(structure \mid sequence).$$

Ce score, au final assez complexe, fait intervenir :

- un terme de compatibilité résidu/environnement,
- un terme d'interaction entre résidus pour favoriser l'enfouissement des résidus hydrophobes et optimiser les interactions entre segments de structures secondaires,
- la compacité de la structure,
- un terme spécifique pour la formation des feuillets β.

Choix du meilleur modèle De 10 000 à 400 000 simulations indépendantes sont effectuées pour chaque séquence cible [30]. Les mauvaises structures sont éliminées par différents filtres qui détectent les conformations dont les caractéristiques sont trop éloignées des structures natives.

Les modèles restants sont regroupés sur des bases structurales. Les séquences homologues de la cible sont également modélisées et regroupées. Le choix final du modèle repose sur l'hypothèse que le bassin d'énergie de la structure native est le plus large, et que c'est donc au voisinage de la structure native que l'on devrait trouver le plus de modèles générés. Les centres des plus grosses classes sont donc proposés comme modèle global de la prédiction.

La méthode Rosetta a donné lieu à des développements méthodologiques, par exemple pour la prédiction de structures d'après les données RMN [29] et le design de nouvelles protéines [48].

1.3 Conclusion

Ce premier chapitre a permis de mettre en place le cadre général de cette thèse : les protéines, les différents niveaux de description de leurs structures et les stratégies existantes pour prédire la structure 3D d'après la séquence.

Les structures de protéines peuvent se révéler, dans certains cas, difficiles à décrire en terme de structures secondaires. Les structures secondaires réelles peuvent être assez différentes des modèles théoriques idéaux. Le prochain chapitre porte sur cet aspect descriptif :

il présente la validation d'une nouvelle méthode d'assignation des structures secondaires qui permet de décrire au mieux les hélices et brins coudées.

La prédiction de structure sera abordée dans les chapitres suivants.

Chapitre 2

Une nouvelle méthode d'assignation des structures secondaires des protéines

La description des structures 3D en terme de structures secondaires est utilisée dans un grand nombre d'applications bio-informatique. Plusieurs logiciels sont actuellement disponibles pour définir automatiquement la position des hélices et des feuillets à partir des structures 3D, comme le programme DSSP [91]. L'existence de ces programmes et leur utilisation courante ne doivent pas faire oublier que l'assignation automatique des structures secondaires peut, dans certains cas, être assez ambiguë et qu'il n'existe malheureusement pas de *bonne réponse*. Nous avons mis au point un nouveau programme d'assignation, appelé KAKSI [1], qui utilise les distances entre Cα et les angles dièdres Φ/Ψ ainsi que des critères spécifiques pour détecter les interruptions dans les hélices α. Pour vérifier la validité de nos assignations, celles-ci ont été comparées aux résultats de plusieurs méthodes, à la fois par des critères globaux et par une analyse détaillée de la géométrie des hélices par un logiciel externe. Ces analyses ont montré que KAKSI permet d'assigner des hélices α relativement linéaires et permet, dans certains cas difficiles, de fournir des assignations plus satisfaisantes que les logiciels les plus utilisés.

2.1 Contexte

La divergence entre la structure secondaire prédite d'une protéine et sa structure secondaire réelle a constitué un sujet d'étude dès l'apparition des premières méthodes de prédiction [174]. Il a fallu plus de temps à la communauté des structuralistes pour prendre conscience du fait que la **détermination** objective des structure secondaires d'après les structures tridimensionnelles n'est pas une tâche triviale. Comme le font remarquer Robson et Garnier dans un ouvrage paru en 1986 [155] : « In looking at a model of a protein, it is often easy to recognize helix and to a lesser extent sheet strands, but it is not easy to say whether the residues at the ends of these features be included in them or not. [...] In addition there are many distorsions within such structures, so that it is difficult to assess whether this represents merely a distortion, or a break in the structure. [...] In fact the problem is essentially that helices and sheets in globular proteins lack the regularity and clear definition found in the Pauling and Corey models. »Ainsi, la plupart des hélices α des protéines globulaires sont légèrement courbées, comme en témoignent les travaux

[1] KAKSI Ce mot signifie *deux* en finnois, référence discrète à la nature *secondaire* des motifs qui nous intéressent.

de Barlow et Thornton [15] et Kumar and Bansal [112, 114]. En conséquence, un groupe d'experts (spectroscopistes RMN et cristallographes) fournirait probablement différentes assignations à partir d'une même structure 3D.

Pour pallier ce problème et analyser automatiquement le nombre grandissant de structures résolues expérimentalement, des programmes d'**assignation automatique** sont développés depuis les années 70. L'objectif des ces logiciels est d'incorporer l'expertise humaine pour fournir des assignations cohérentes et reproductibles. L'existence des structures secondaires régulières (hélices α, feuillets β) peut être mise en évidence grâce aux liaisons hydrogènes qui les stabilisent, mais d'autres critères peuvent être utilisés. En effet, les structures secondaires régulières sont des motifs périodiques, elles génèrent donc des régularités dans la géométrie de la chaîne principale. Ainsi les distances entre Cα ou les angles α (angles dièdres formés par la succession de quatre Cα) peuvent aussi être permettre de caractériser la présence des hélices et des feuillets. Les angles dièdres Φ/Ψ ont des valeurs caractéristiques dans les hélices α et les feuillets β, comme le montre la figure 2.1. La représentation en densités montre que la répartition des angles dièdres dans les hélices est très concentrée autour d'un seul pic alors que la variabilité est beaucoup plus grande dans les feuillets β.

La première implémentation d'une méthode d'assignation automatique remonte à 1977, par Levitt et Greer [118]. L'algorithme utilisé reposait principalement sur les angles de torsion entre Cα.

Quelques années plus tard, Kabsh et Sander développent DSSP [91] (*Description of Secondary Structure of proteins*), qui reste à l'heure actuelle le programme le plus couramment utilisé. L'algorithme de DSSP est basé sur la détection des liaisons hydrogènes, définies selon un critère électrostatique. L'assignation des structures secondaires est ensuite réalisée en recherchant des arrangements caractéristiques de liaisons hydrogènes. Cette méthode a rapidement été acceptée comme standard, c'est pourquoi de nombreux logiciels l'utilisentRASMOL [167], l'un des logiciels de visualisation les plus distribués, assigne les structures secondaires régulières grâce à un algorithme rapide dérivé de DSSP. Les outils d'analyse du logiciel GROMACS utilisent également DSSP [20].

STRIDE [69] est un logiciel ressemblant à DSSP. Les liaisons hydrogènes y sont utilisées comme dans l'algorithme de DSSP, bien que leur définition soit légèrement différente. De plus, STRIDE tient compte des valeurs des angles (Φ/Ψ). STRIDE est utilisé par le logiciel

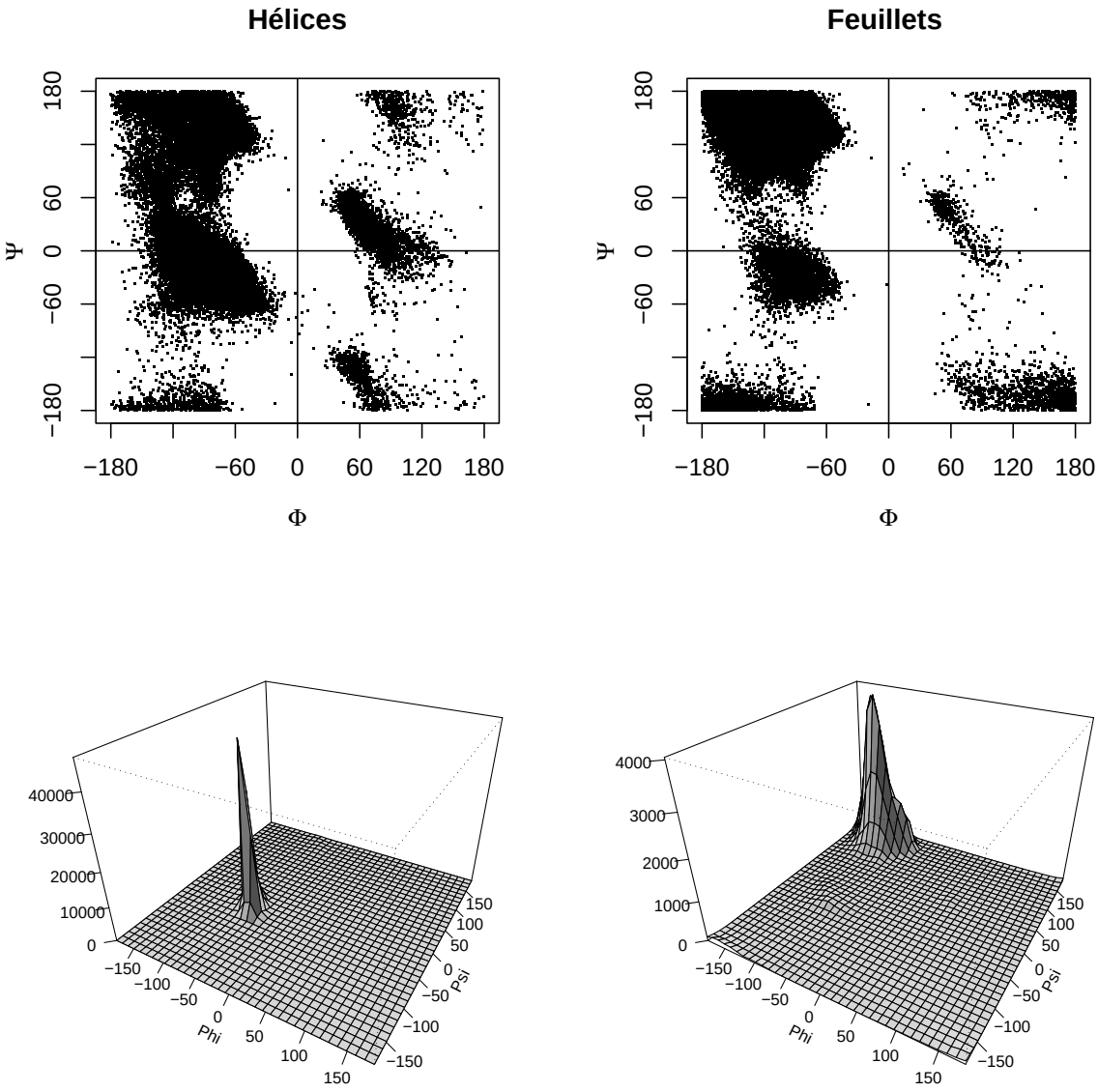

FIG. 2.1 – Répartition des angles dièdres Φ/Ψ dans les hélices α et les feuillets β, en représentation usuelle et en densités. Les données sont extraites de 2505 structures non redondantes de la PDB. Les définitions des hélices et des feuillets sont tirées des fichiers PDB.

de visualisation VMD [85].

SECSTR [67] appartient à la même famille de méthodes. Ce logiciel a été développé pour améliorer la détection des hélices π. En effet, les auteurs de SECSTR avaient noté que DSSP et STRIDE ne permettaient pas de détecter plusieurs hélices π qu'ils avaient caractérisées manuellement.

D'autres méthodes ont été développées, utilisant d'autres critères. DEFINE [151] repose sur les coordonnées des Cα et compare les distances entre Cα aux distances dans des segments de structures secondaires idéaux. DEFINE fournit également une description en terme de structures super-secondaires. L'approche implémentée dans le logiciel P-CURVE [182] est basée sur une définition des paramètres hélicoïdaux des peptides et la génération d'axes globaux des peptides. PSEA [115] utilise les coordonnées des Cα seuls. L'algorithme est basé sur des critères d'angle et de distance. XTLSSTR [105] a été développé dans le but d'assigner les structures secondaires *à la manière d'un expert*, d'après des distances et des angles calculés sur le squelette de la protéine. Les méthodes les plus récentes, à notre connaissance, utilisent les polygones de Voronoï (méthode VOTAP) [55] et la triangulation de Delaunay [188].

Les fichiers de structures fournis par la Protein Data Bank contiennent une description de la structure secondaires des protéines déposées, dans les champs HELIX, SHEET et TURN[2]. Ces descriptions peuvent être fournies par le dépositaire (de manière optionnelle) ou générées automatiquement par DSSP. Environ 90% des fichiers PDB contiennent de telles descriptions. Cependant, lorsqu'elles existent, il arrive que ces descriptions soient partielles.

La diversité des méthodes existantes illustre bien le fait qu'il existe plusieurs façons légitimes de définir les structures secondaires. Il n'est pas surprenant que ces différentes méthodes produisent des assignations légèrement divergentes, particulièrement aux extrémités des hélices et des brins. Colloc'h et al ont ainsi montré que le pourcentage d'accord entre DSSP, P-CURVE et DEFINE n'est que de 63% et que DEFINE tend à sur-assigner les structures secondaires régulières [44]. Les auteurs de XTLSSTR ont remarqué que DSSP assigne plus de brins β que XTLSSTR [105]. SECSTR est logiquement plus sensible dans la détection des hélices π que DSSP ou STRIDE [67]. Les figures 2.2, 2.3 et 2.4 montrent quelques exemples de structures tridimensionnelles pour lesquelles les assignations fournies par DSSP, STRIDE, PSEA, XTLSSTR et SECSTR sont sensiblement différentes. Les différences portent principalement sur les extrémités des hélices et des brins, la présence ou non d'hélices et de brins très courts et l'assignation de longs brins déformés qui sont décrits comme plusieurs brins par certaines méthodes.

[2]voir PDB Format Description Version 2.2, `http://www.rcsb.org/pdb/docs/format/pdbguide2.2/guide2.2_frame.html`

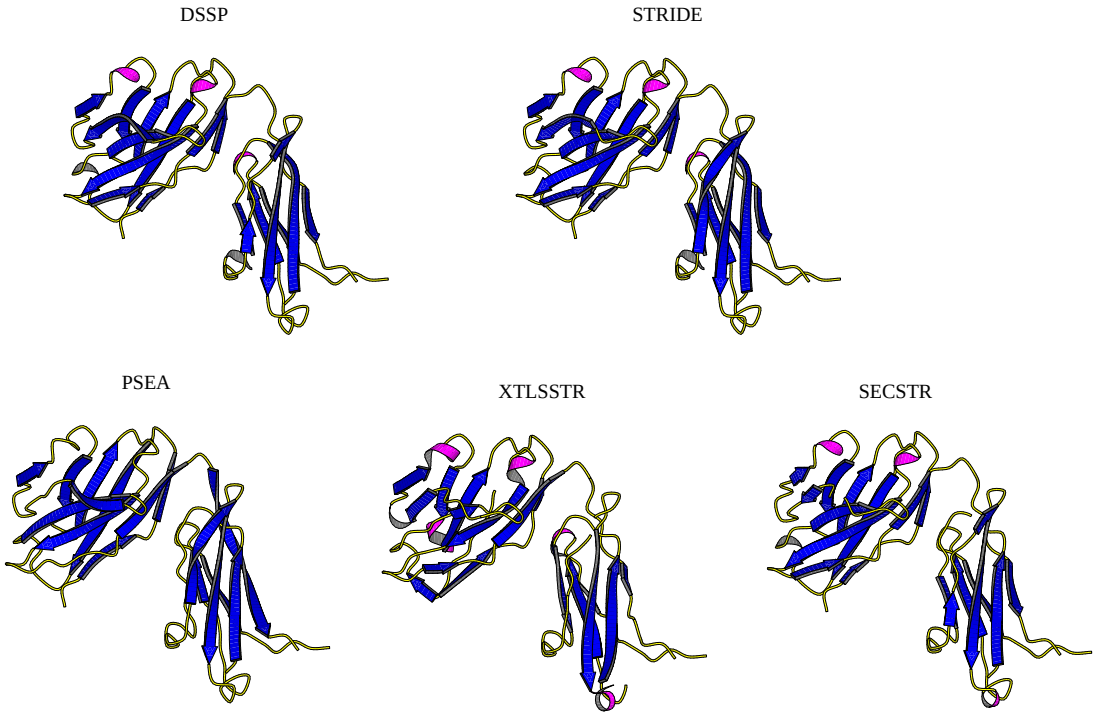

FIG. 2.2 – Structure d'un fragment d'anticorps de souris (code PDB 1mju, chaîne H, résolution 1.22 Å) décrite par différents programmes d'assignation. Image générées par molscript [107].

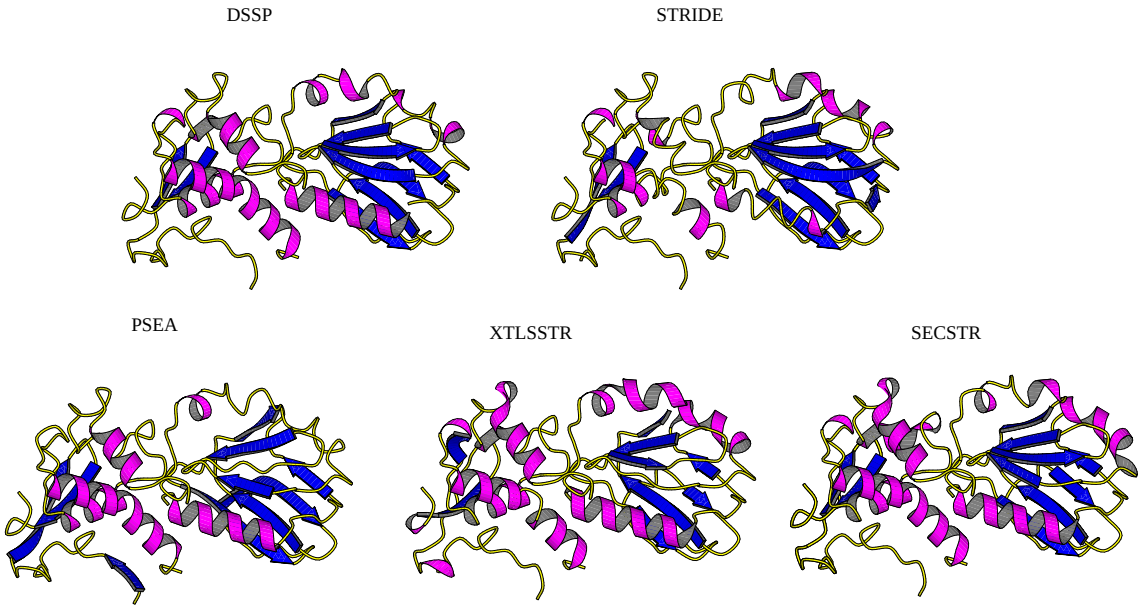

FIG. 2.3 – Structure d'une endonucléase d'Escherichia Coli (code PDB 1k3x, chaîne A, résolution 1.25 Å) décrite par différents programmes d'assignation.

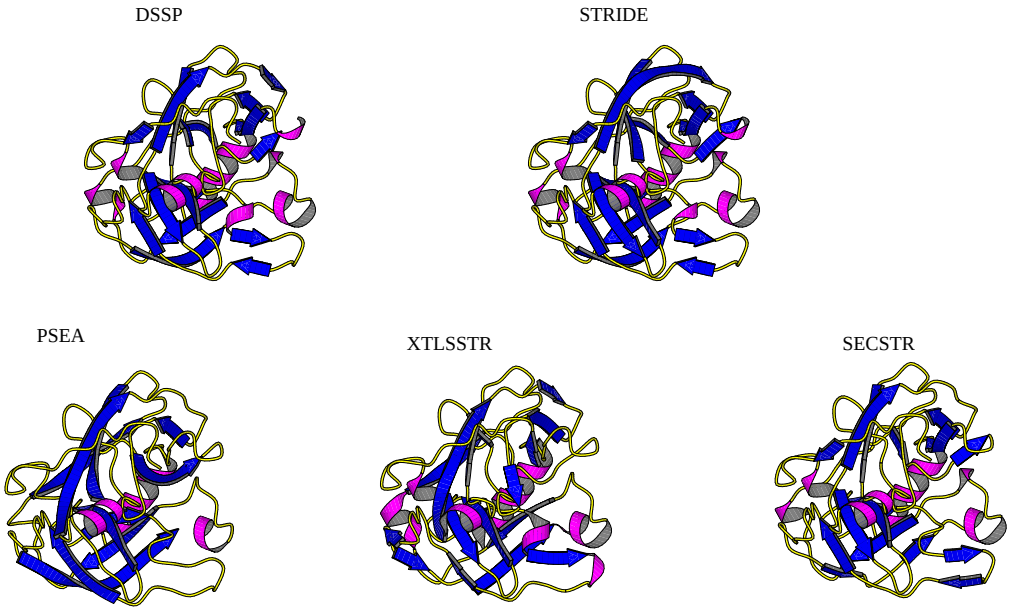

FIG. 2.4 – Structure d'une endopeptidase de Bacillus intermedius (code PDB 1p3c, chaîne A, résolution 1.5 Å) décrite par différents programmes d'assignation.

Dans la suite de ce chapitre, nous présentons une nouvelle méthode d'assignation appelée KAKSI, basée sur les distances entre Cα et les angles (Φ/Ψ). Ces caractéristiques sont intuitivement utilisées lors de l'examen visuel des structures. Notre objectif, en développant cette méthode, est de traiter de manière satisfaisante les problèmes d'irrégularités dans les structures secondaires régulières. Nous avons considéré que les régions de la chaîne polypeptide montrant un changement abrupt d'orientation (comme les coudes dans les hélices) doivent être considérées comme une interruption dans la structure secondaire régulière. Une méthode d'assignation doit fournir des assignations fiables et correctes. Il est extrêmement difficile de prouver que notre méthode représente une amélioration par rapport aux méthodes existantes, en raison de l'absence de standard de référence pour l'évaluation. Nous avons donc mené une étude comparative des assignations fournies par KAKSI et plusieurs autres méthodes basées sur différents critères : DSSP, STRIDE, SECSTR, XTLSSTR, PSEA, ainsi que les description fournies par la PDB. Il est connu que la résolution des structures résolues par cristallographie aux rayons X a un effet direct sur la qualité des structures déposées. La détection des structures secondaires peut s'avérer plus difficile sur les structures de basse résolution [69]. Il est intéressant, dans cette optique,

d'évaluer l'effet de la résolution sur l'assignation des structures secondaires par différentes méthodes. Les structures résolues par RMN correspondent à des protéines en solution et fournissent une représentation plus *dynamique* de la conformation des protéines que les structures obtenues par cristallographie. Les structures RMN sont donc plus sujettes aux distorsions locales et constituent un cas difficile et intéressant pour les méthodes d'assignation automatique. Les comparaisons sont donc effectuées sur 4 jeux de protéines : 3 jeux de structures cristallographiques de résolution haute, moyenne et basse, ainsi qu'un jeu de structures RMN. Ces différents jeux de données nous permettent d'évaluer l'effet de la résolution et de la méthode expérimentale sur les différentes méthodes d'assignation. La question de l'inclusion des résidus terminaux dans la définition des hélices et des brins est abordée par l'examen des longueurs des segments assignés par les différents algorithmes. Les distorsions des structures secondaires sont aussi un objet d'étude. Plus spécifiquement, la géométrie des hélices est analysée par HELANAL [14], un logiciel dédié à la caractérisation de la géométrie des hélices. Pour finir, quelques exemples d'assignation difficile sont présentés, avec des distorsions dans les structures secondaires, et des assignation divergentes fournies par KAKSI et STRIDE.

2.2 Matériel et méthodes

2.2.1 Données

La méthode implémentée dans KAKSI utilise un certain nombre de valeurs caractéristiques décrivant la géométrie des hélices α et des feuillets β. Ces valeurs caractéristiques sont extraites des structures disponibles. Un jeu de protéines de référence (*Ref set*), constitué de 2880 domaines structuraux définis par la classification ASTRAL 1.63 [31] est utilisé pour extraire ces valeurs caractéristiques. Une liste de domaines avec moins de 40% d'identité de séquence a été récupéré sur le serveur ASTRAL[3]. Seules les structures résolues avec une résolution inférieure à 2.25 Å et longues d'au moins 50 résidus sont retenues.

Les assignations fournies par KAKSI sont ensuite comparées avec les assignations réalisées par les autres méthodes. Pour les raisons mentionnées précédemment, nous utilisons 4 jeux de données de comparaisons. Les tailles des jeux de données correspondent aux

[3] `http://astral.berkeley.edu/`

fichiers effectivement traités par tous les programmes d'assignation et contenant une description des structures secondaires (champs HELIX et SHEET).

Un lot de structures cristallographiques de haute résolution (*HRes set*) : de résolution inférieure à 1.7 Å, avec un facteur R inférieur à 0.19, et moins de 30% d'identité entre séquences. Cette liste de structures est obtenue sur le site web WHATHIF [82] [4]. Ce jeu de données contient 689 structures, ce qui correspond à 151 922 résidus de structure secondaire définie (à l'exclusion des coordonnées manquantes).

Un jeu de structures cristallographiques de moyenne résolution (*MRes set)* : de résolution comprise entre 1.7 Å et 3 Å, facteur R inférieur à 0.3, et moins de 30% d'identité entre séquences. La liste des structures est obtenue sur le site web PISCES [197] [5]. Ce jeu de données contient 624 structures, ce qui correspond à 160 276 résidus de structure secondaire définie.

Un jeu de structures cristallographiques de basse résolution (*LRes set*) : de résolution supérieure à 3 Å, avec un facteur R supérieur à 0.3, et moins de 30% d'identité entre séquences. La liste des structures est obtenue sur le site web PAPIA [6]. Ce jeu de données contient 332 structures, ce qui correspond à 97 852 résidus de structure secondaire définie.

Un jeu de structures RMN (*RMN set*) : avec moins de 30% d'identité entre séquences. La liste des structures RMN est obtenue sur le site web de la PDB [7]. La redondance en séquences est réduite à 30% d'identité avec PISCES. Ce jeu de données contient 296 structures, ce qui correspond à 27 533 résidus de structure secondaire définie.

2.2.2 Nouvelle méthode d'assignation : KAKSI

L'assignation des structures secondaires répétitives par KAKSI utilise des valeurs de distances entre Cα et d'angles dièdres (Φ/Ψ) caractéristiques des hélices et des feuillets. Pour obtenir ces valeurs et paramétrer notre méthode, nous utilisons les descriptions de structures secondaires contenues dans les fichiers PDB (champs HELIX et SHEET). Ces champs, lorsqu'ils sont présents, sont remplis en utilisant DSSP ou la description fournie par le dépositaire de la séquence (qui peut avoir utilisé un algorithme d'assignation). Pour

[4] `http://swift.cmbi.kun.nl/whatif/select/`

[5] `http://dunbrack.fccc.edu/PISCES.php`

[6] `http://mbs.cbrc.jp/papia/papia.html`

[7] `http://www.rcsb.org/pdb/`

la mise en place de notre méthode, ces descriptions seront considérées comme standard de référence, malgré le biais probable vers DSSP.

L'assignation est réalisée dans une fenêtre glissante le long de la séquence. Les hélices α sont assignées en premier, puis les feuillets β, au moyen de deux fenêtres glissantes. Les résidus assignés en hélices α ne peuvent pas être réassignés en feuillet.

Caractéristiques géométriques des structures secondaires régulières utilisées par kaksi

Les hélices α et les feuillets β étant des structures périodiques, leur présence se traduit par une régularité dans la géométrie du squelette de la protéine, ce qui génère des distances particulières entre Cα et des valeurs particulières des angles dièdres (Φ/Ψ).

Plus précisément, nous avons extrait, à partir du *Ref set* :

- les distances caractéristiques entre Cα dans les hélices α et les feuillets β,
- les répartitions des angles dièdres (Φ/Ψ) caractéristiques des hélices α et des brins β.

Distances inter Cα. Dans les hélices α, 4 distances sont considérées entre les résidus i et j le long de la séquence, avec $j \in [i+2, i+5]$. Nous rapportons, dans le tableau 2.1 les moyennes et écarts-types obtenus sur le *Ref set*. Les trois distances mesurées dans les feuillets β sont illustrées et rapportées dans la figure 2.5. Nous distinguons pour certaines distances, les distances incluant les résidus terminaux des segments.

TAB. 2.1 – Distances dans les hélices α

Type	Coeur[a]	Extrémités [b]
$i/i+2$	5.49(0.20) [c]	5.54(0.25)
$i/i+3$	5.30(0.64)	5.36(0.39)
$i/i+4$	6.33(0.71)	
$i/i+5$	8.72(0.63)	

[a]distances calculées dans les coeurs d'hélices.
[b]distances impliquant des résidus terminaux.
[c]Les distances moyennes sont indiquées en Å avec leurs écarts-types entre parenthèses.

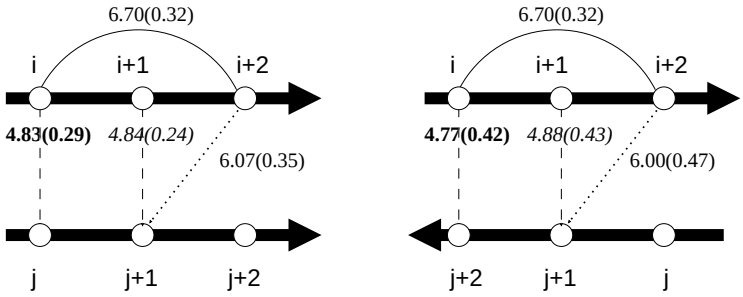

FIG. 2.5 – Distances mesurées dans les feuillets β parallèles (à gauche) et antiparallèles (à droite). Les moyennes en Å calculées sur le *Refset* sont indiquées avec leurs écarts-types entre parenthèses. Les distances en italiques sont calculées dans les coeurs, les distances en gras impliquent des résidus terminaux. La distinction d'orientation (parallèle/antiparallèle) n'est pas faite pour la distance intra-brin (type i - $i+2$).

Les répartitions des angles Φ/Ψ sont prises compte en utilisant des cartes de Ramachandran. Les cartes sont discrétisées en carrés de 10 degrés de côté. Nous générons ainsi deux *cartes de population*, l'une spécifique des résidus en hélice, l'autre spécifique des résidus en feuillet, qui contiennent les comptages observés sur le *Ref set*. Les comptages sont normalisés. Les deux cartes de population sont visibles dans la figure 2.6.

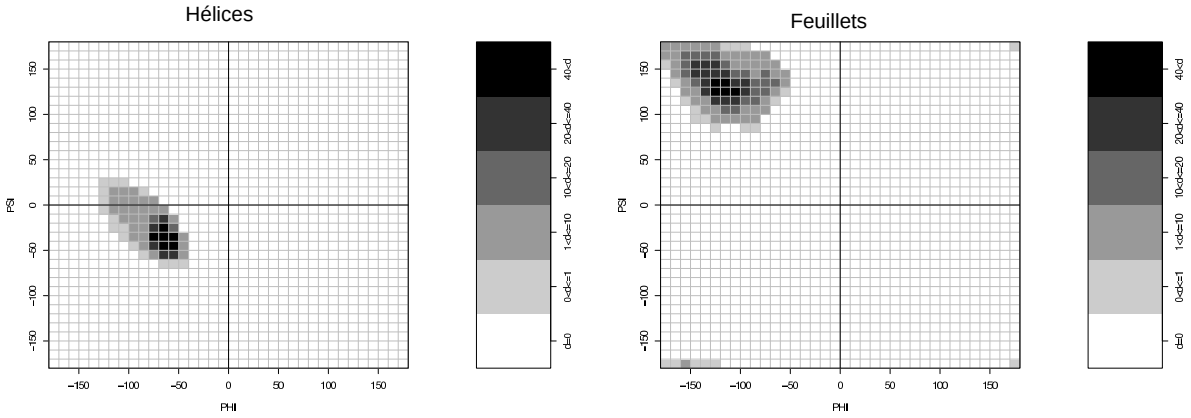

FIG. 2.6 – Cartes de population des hélices α et des feuillets β.

Pour l'utilisation dans l'algorithme d'assignation, les comptages sont normalisés par le nombre moyen d'observations par carré. Dans la carte des hélices α, les comptages hors de l'aire ($\Phi < 0°$ et $-90° < \Psi < 60°$) sont mis à zéro. Lors de l'assignation, les angles dièdres doivent correspondre à une zone de la carte avec un comptage normalisé supérieur à un seuil δ_H. Dans cette étude, δ_H est fixé à 20.

Un soin particulier a été apporté à la détection des coudes dans les hélices α. De nom-

breuses hélices sont coudées, mais les critères de distances et d'angles dièdres ne suffisent pas toujours pour détecter ces déformations. C'est pourquoi deux critères spécifiques - décrits dans la section suivante- sont mis en place pour détecter les coudes après l'étape d'assignation. L'un deux utilise les distances entre paires d'angles dièdres. Dans une hélice α parfaitement régulière, les valeurs d'angles dièdres sont très proches. Les distances entre paires d'angles dièdres, représentées dans la figure 2.7, devraient donc être très petites.

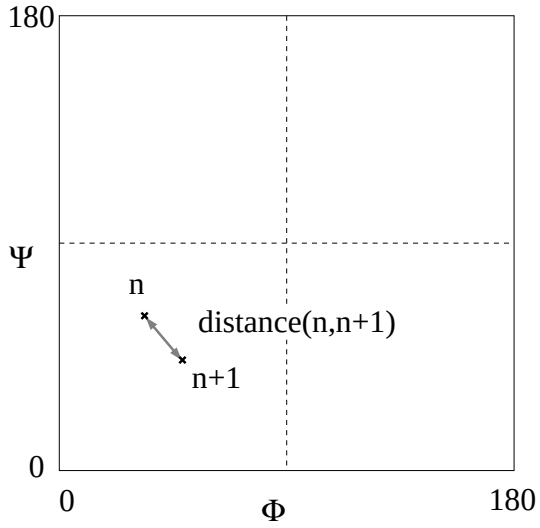

FIG. 2.7 – Distance entre deux paires d'angles dièdres dans le diagramme de Ramachandran.

C'est cette propriété qui est utilisée pour détecter un coude : la distance entre deux paires d'angles dans le diagramme de Ramachandran est comparée à la distribution des distances dans le *Ref set*.

Heuristique d'assignation implémentée dans kaksi

La procédure d'assignation est résumée dans la figure 2.8. Cette approche a été retenue après avoir testé plusieurs critères et plusieurs façons de combiner les critères, car elle génère des assignations en bon accord avec les structures décrites dans les fichiers PDB et satisfaisantes lors de l'examen visuel des structures. Le principe de l'assignation est de tester les distances entre Cα le long de la protéine pour en vérifier l'adéquation avec les distances typiques attendues dans les structures secondaires régulières. Les valeurs des angles dièdres (Φ/Ψ) sont testées de la même manière. L'assignation des hélices est réalisée par la satisfaction d'un critère de distance **ou** d'angle. La détection des feuillets β requiert

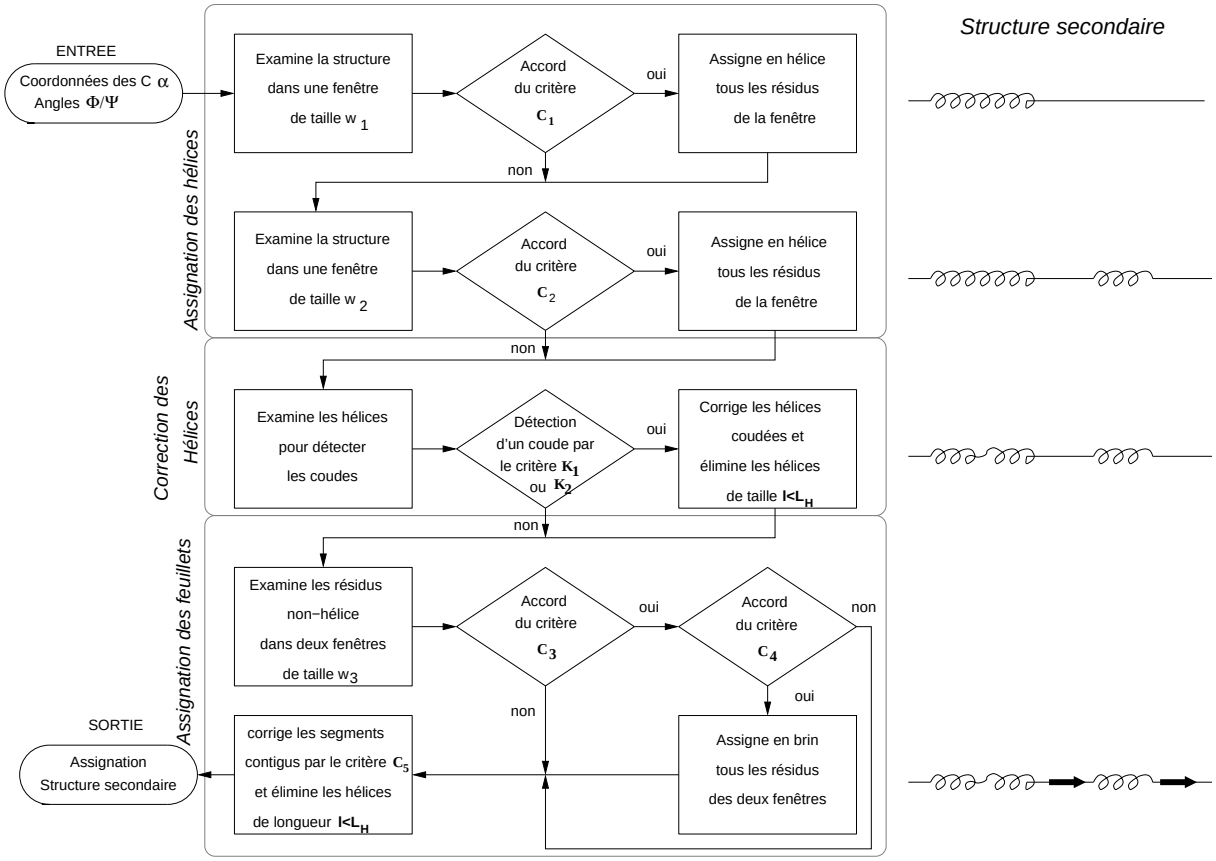

FIG. 2.8 – Procédure de détection des structures secondaires régulières implémentée dans KAKSI. La longueur minimale des hélices est fixée à $L_H = 5$. Les critères C1, C2, C3, C4, C5, K1 et K2 sont expliqués dans le texte.

la satisfaction des critères de distance **et** d'angle. Les assignations des hélices sont ensuite corrigées par la détection des coudes. Les critères appliqués à chaque étape de la procédure présentée dans la figure 2.8 sont détaillés ci-dessous dans l'ordre de leur utilisation. Les valeurs typiques d'angles et de distances extraites de l'ensemble *Ref set* sont symbolisées par des noms en capitale. Les paramètres internes de la méthode d'assignation sont les suivants : ε_H et ε_b définissent des seuils pour les distances entre Cα distances, η_H et σ_b définissent des seuils pour les valeurs des angles dièdres (Φ/Ψ).

Critère de distance pour les hélices α (C1). Toutes les distances entre Cα dans une fenêtre glissante de taille w_1 (fixée à 6) doivent être dans l'intervalle $[M_\alpha - \varepsilon_H \times SD_\alpha; M_\alpha + \varepsilon_H \times SD_\alpha]$. M_α et SD_α représentent la moyenne et l'écart-type des distributions de distances dans les hélices α, estimés à partir du *Ref set*

Critère d'angle pour les hélices α (C2). Toutes les paires (Φ/Ψ) dans une fenêtre glissante de longueur w_2 (fixée à 4 dans cette étude) doivent satisfaire la condition ($\Phi < 0°$

et $-90° < \Psi < 60°$) et une paire, au moins, doit être située dans la zone de la carte de population avec une densité supérieure à $> \delta_H$, δ_H étant fixé à 20.

Détection des coudes dans les hélices à l'aide de deux critères (K1 et K2).

• **Le critère K1** est basé sur les valeurs des angles dièdres (Φ/Ψ). Une hélice est interrompue au résidu $j+1$ si la somme $d_{\Phi/\Psi}(j, j+1) + d_{\Phi/\Psi}(j+1, j+2)$ est supérieure à $\eta_h \times D^{95}_{\Phi/\Psi}$. $d_{\Phi/\Psi}(j, j+1)$ est une mesure de la distance entre les paires d'angles associée aux résidus j et $j+1$ dans la carte de Ramachandran, analogue à la déviation moyenne angulaire décrite par Shuchhardt et collaborateurs [173]. $D^{95}_{\Phi/\Psi}$ est le percentile 95 de la distribution de distances, estimé d'après le *Ref set*.

• **Le critère K2** se fonde sur des axes. L'axe d'une hélice est calculé en minimisant la fonction $D_{axis} = \frac{1}{n}\sum_i (d_i - d_m)^2$ où n est le nombre de résidus dans l'hélice, d_i est la distance du *ième* Cα à l'axe et d_m est la moyenne des d_i. Pour une hélice parfaitement linéaire, D_{axis} vaut zéro et le vecteur correspondant est l'axe du cylindre circonscrit par les Cα. Une hélice est interrompue si l'ajustement est meilleur avec deux axes formant un angle supérieur à θ_k (ici, $\theta_k = 25°$).

Critère de distance pour les feuillets β (C3). Toutes les distances entre Cα dans deux fenêtres glissantes de longueur w_3 (ici $w_3 = 3$) doivent être dans l'intervalle $[M_\beta - \varepsilon_b \times SD_\beta ; M_\beta + \varepsilon_b \times SD_\beta]$. M_β et SD_β sont la moyenne et l'écart-type des distributions de distances, calculés sur le *Ref set*.

Critère d'angle pour les feuillets β (C4). Pour chaque paire d'angles (Φ/Ψ) tombant dans une zone de la carte de population ayant une densité supérieure à zéro, un compteur $score(feuillet)$ est incrémenté de 1. Si la paire d'angle (Φ/Ψ) d'un résidu central d'une fenêtre vérifie $-120° < \Psi < 50°$, alors $score(feuillet)$ est remis à zéro. $score(feuillet)$ doit être supérieur ou égal à σ_b.

Correction des segments consécutifs (C5). Si une hélice et un brin sont adjacents, l'hélice est raccourcie d'un résidu par l'introduction d'un coil.

Les valeurs optimales des paramètres ont été empiriquement fixées à : $\varepsilon_H = 1.96$, $\eta_H = 2.25$, $\varepsilon_b = 2.58$ et $\sigma_b = 5$.

2.2.3 Méthodes d'assignation utilisées pour la comparaison

Les assignations produites par KAKSI sont comparées à celles fournies par cinq autres méthodes : DSSP [91], STRIDE [69], PSEA [115], XTLSSTR [105] et SECSTR [67]. Les champs

HELIX et SHEET des fichiers PDB sont considérés comme une méthode d'assignation indépendante.

Les assignations sont converties, si besoin, en trois classes classiques (H pour hélice α, b pour brin β, c pour coil) : DSSP, STRIDE et SECSTR : (H, G, I) = H, (E, b) = b, autres (S, T, espace) = c ; XTLSSTR : (G, g, H, h) = H, (E, e) = b, autres (T, N, P, p, -) = c. PSEA assigne trois états. XTLSSTR peut fournir plusieurs assignations alternatives. Seule la première a été prise en compte. Dans le cas des structures RMN, seul le premier modèle est analysé.

2.2.4 Indices globaux de comparaison

Composition en structures secondaires

Le contenu en structures secondaires est décrit par le pourcentage de résidus impliqués dans chacune des trois classes structurales : hélice α, brin β et coil.

Accord global entre deux méthodes

Le score C_3 est le pourcentage de résidus assignés dans le même état par deux méthodes d'assignation : $C_3 = N_{id}/N_{tot}$ avec N_{id} le nombre de résidus assignés à l'identique par les deux méthodes et N_{tot} le nombre total de résidus assignés. Ce score est analogue au Q_3 utilisé pour évaluer les méthodes de prédiction de structure secondaire.

Mesure de recouvrement des segments

Le recouvrement des segments de structures secondaires est mesuré par le score SOV (Segment OVerlap), défini par Zemla et collaborateurs [208]. Pour un état structural i (hélice α, brin β ou coil) le SOV est donné par :

$$SOV(i) = \frac{1}{N(i)} \sum_{s(i)} \frac{minov(s_1, s_2) + delta(s_1, s_2)}{maxov(s_1, s_2)} \times len(s_1)$$

avec $N(i)$ défini par :

$$N(i) = \sum_{S(i)} len(s_1) + \sum_{S'(i)} len(s_1).$$

Les sommes sur les $S(i)$ incluent toutes les paires de segments i ayant au moins un résidu de recouvrement. Les sommes sur les $S'(i)$ incluent les segments de l'assignation

1 ne donnant pas lieu à des paires ; $len(s_1)$ est le nombre de résidus du segment s_1, $minov(s_1, s_2)$ la longueur du recouvrement entre s_1 et s_2 ; $maxov(s_1, s_2)$ est l'étendue totale recouverte par les deux segments ; $delta(s_1, s_2)$ est défini par :

$$min\left\{maxov(s_1, s_2) - minov(s_1, s_2); minov(s_1, s_2); int(\frac{len(s_1)}{2}); int(\frac{len(s_2)}{2})\right\},$$

avec $min\{x1; x2; x3; ...; xn\}$ le minimum de n entiers et $int(x)$ la partie entière de x.

Le SOV s'utilise habituellement pour évaluer une prédiction de structures secondaires (S_2) par rapport à la vraie structure (S_1). S_1 et S_2 n'ont donc pas un rôle symétrique.

D'autre part nous avons étudié les paires de longueurs de segments utilisées pour le calcul du SOV. Une paire est définie par au moins un résidu assigné dans le même état par les deux méthodes. Ces paires de longueurs peuvent être visualisées sous forme d'un bi-plot(longueur(X) *vs* longueur(Y)). Les segments ne donnant pas lieu à des paires ne sont pas considérés dans cette analyse.

2.2.5 Analyse de la géométrie des hélices avec un logiciel externe

Le logiciel HELANAL développé par Kumar and Bansal [14] est dédié à la caractérisation de la géométrie des hélices d'au moins 9 résidus. HELANAL prend en entrée un fichier PDB et une description des limites des hélices. Le logiciel calcule des axes locaux pour 4 résidus. La géométrie d'une hélice est déterminée par les angles entre axes locaux et l'ajustement de la trace de l'hélice avec un cercle ou une droite. Les hélices sont finalement classifiées comme coudées (K), linaires (L) ou courbées (C). La figure 2.9 présente des exemples de ces trois types d'hélices.

Si la géométrie d'une hélice est ambiguë, elle peut ne pas être classifiée par le programme. Dans cette étude, HELANAL est utilisé comme contrôle externe de la géométrie des hélices. Toutes les hélices α de plus de 9 résidus sont soumises à l'analyse par HELANAL. Les différentes méthodes d'assignation sont utilisées pour proposer des limites alternatives d'hélices.

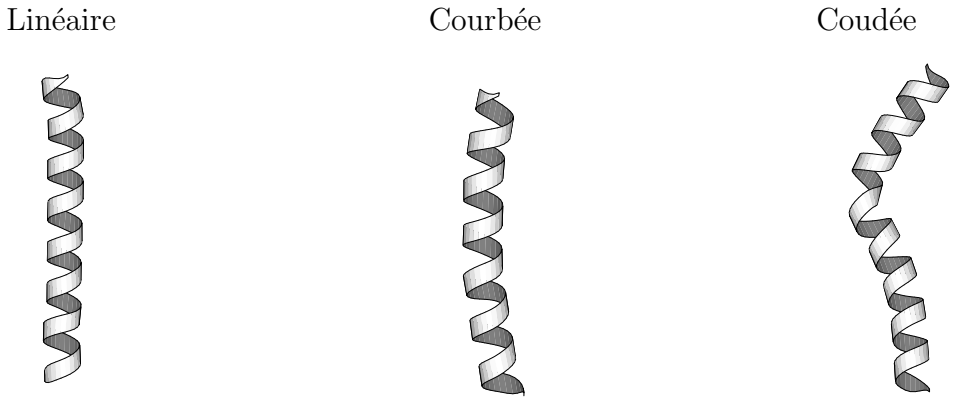

FIG. 2.9 – Exemples d'hélices α linéaire (structure 1rib, chaîne A, résidus 102-129), courbée (structure 1bge, chaîne B, résidus 144-169) et coudée (structure 1lis, résidus 44-74), d'après [14].

2.3 Résultats et discussion

2.3.1 Paramètres internes de kaksi

La détection des structures secondaires par KAKSI dépend de plusieurs paramètres. Pour tester la robustesse de la méthode envers ces paramètres, nous avons analysé l'effet du changement de ε_H, ε_b et σ_b sur les compositions en structures secondaires des jeux de comparaisons. ε_H et ε_b varient entre 1.29 et 3.30, σ_b entre 3 et 6. L'effet de chaque paramètre est testé **séparément**, les autres paramètres gardant leurs valeurs optimales.

TAB. 2.2 – Effet des paramètres ε_H, ε_H et σ_b sur le contenu en structures secondaires des assignations par KAKSI sur les structures du *HRes set*.

ε_H	%H[a]	%b[b]	ε_b	%H	%b	σ_b	%H	%b
1.29	36.10	22.00	1.29	36.10	6.94	3	36.48	23.96
1.65	36.56	22.00	1.65	36.56	14.52	4	36.57	23.59
1.96	36.82	22.00	1.96	36.82	18.55	5	36.82	22.00
2.58	37.11	22.00	2.58	37.11	22.00	6	37.27	17.51
3.30	37.32	22.00	3.30	37.62	23.53			

[a]taux de résidus assignés en hélice
[b]taux de résidus assignés en feuillet

Les effets sont similaires sur les 4 jeux de données. Nous rapportons les résultats obtenus sur le *HRes set* dans le tableau 2.2. La variation du pourcentage d'hélice n'est que modérément sensible aux variations de ε_H : la variation absolue n'excède pas deux

points pour ε_H allant de 1.29 à 3.30 sur le *HRes set*.

La proportion de feuillets β diminue clairement quand ε_b est inférieur à 2.58 et augmente si ε_b est supérieur à 2.58. Le taux de β est légèrement augmenté par une diminution de σ_b et clairement diminué par son augmentation. Les variations des paramètres de détection des feuillets ont un léger effet sur le contenu en hélice. La procédure de correction des segments consécutifs (critère C5) explique cette variation : les feuillets sont prioritaires ; les hélices α sont raccourcies en cas de segments adjacents dans la séquence.

Nous observons donc deux comportements : une détection des hélices α assez peu sensible à ses paramètres de détection et une détection des feuillets β assez sensible à des changements de paramètres. La structure de l'algorithme employé explique ces différences. La détection des hélices est accomplie par la satisfaction d'un critère d'angle ou de distance. Des changements modérés dans l'un des deux critères peuvent être compensés par l'autre critère. Au contraire, la détection des feuillets nécessite la satisfaction des deux critères.

L'efficacité de nos deux critères de détection des coudes a également été testée : K1, le critère utilisant les angles dièdres, et K2, le critère utilisant les axes.

TAB. 2.3 – Effet des critères de détection des coudes sur le taux d'hélices coudées d'après HELANAL sur les structures du *HRes set*.

critère de détection des coudes	%K[a]
aucun	40.26
K1	23.80
K2	28.24
K1+K2	21.95

[a]taux d'hélices de plus de 9 résidus analysées comme coudées par HELANAL

La géométrie des hélices est analysée avec HELANAL et la proportion d'hélices classées coudées est calculée. Ces résultats sont rapportés dans le tableau 2.3 pour le *HRes set*. La proportion d'hélices coudées est diminuée par l'application de chacun des deux critères, pris séparément. Le critère K1 permet la plus grande diminution du taux d'hélices coudées selon HELANAL. Une diminution encore plus grande est obtenue quand les deux critères sont utilisés successivement.

Par la suite, les assignations du programme KAKSI seront produites en utilisant les

paramètres par défaut et les deux critères K1 et K2 pour la détection des coudes dans les hélices.

2.3.2 Sensibilité des méthodes envers la résolution

La composition en structure secondaire permet d'évaluer la sensibilité des méthodes de détection envers la résolution des structures. Le tableau 2.4 rapporte les compositions en structures secondaires des 4 jeux de données de comparaison, assignés par les cinq programmes existants, KAKSI et la description fournie par la PDB.

TAB. 2.4 – Composition en structures secondaires en fonction des méthodes d'assignation

Données	*HRes set*		*MRes set*		*LRes set*		*NMR set*	
Méthode	%H [a]	%b [b]	%H	%b	%H	%b	%H	%b
KAKSI	36.8	22.0	38.0	22.5	35.1	19.0	33.5	15.2
PDB	40.5	20.3	41.7	20.9	39.3	18.2	35.5	17.3
DSSP	35.9	22.5	37.3	22.9	35.4	20.4	32.2	17.3
STRIDE	36.4	22.6	38.6	23.3	36.3	21.2	33.7	18.8
PSEA	32.1	23.7	34.2	25.0	33.0	24.4	30.6	22.8
SECSTR	37.2	20.1	38.5	20.4	37.0	18.6	33.3	16.3
XTLSSTR	40.4	19.7	40.9	19.6	35.9	14.4	34.3	14.8

[a] pourcentage de résidus assignés en hélice α
[b] pourcentage de résidus assignés en brin β

Il n'y a pas de consensus strict entre les différentes méthodes d'assignation concernant la composition en structures secondaires, même sur des structures de haute résolution. Ces variations illustrent la divergence des définitions utilisées. STRIDE et DSSP donnent des compositions très voisines, comme attendu étant donnée la similarité entre ces deux méthodes. PSEA assigne de façon systématique moins de résidus en hélice mais plus de résidus en brin que les autres méthodes. Les descriptions de la PDB sont plus riches en hélice α que les assignations générées automatiquement par les programmes. Notre méthode KAKSI assigne des taux de structures secondaires régulières comparables à STRIDE et DSSP sur le *HRes set*.

Les compositions sont comparables entre les structures *HRes* et les structures *MRes* avec une méthode donnée. En revanche celles des jeux de données *LRes* et *NMR sets* montrent une baisse du taux de structures secondaires régulières. Cette baisse est visible

quelque soit la méthode d'assignation utilisée bien que dans des proportions différentes. En particulier, peu de brins β sont assignés sur le jeu de données *LRes set* par la majorité des méthodes. Seul PSEA donne un taux de β constant quelque soit la résolution des structures. Globalement, l'impact de la résolution sur la détection des structures secondaires par les algorithmes d'assignation est modéré. Le type de technique utilisé (rayons X *vs* RMN) semble avoir un effet plus prononcé.

La diminution de la détection des feuillets β par notre méthode sur les structures *LRes* et RMN indique que des paramètres de détection moins stricts sont plus adaptés pour l'analyse de ces structures. Par exemple, avec $\sigma_b = 3$ (au lieu de 5), le taux de β sur le *LRes set* est de 22.3%, et de 20.7% si l'on choisit $\varepsilon_b = 3.30$ (au lieu de 2.58). De la même manière, le taux de β est de 17.7% avec $\sigma_b = 3$ ou avec $\varepsilon_b = 3.30$ sur les structures du *RMN set*.

2.3.3 Mesure de l'accord global entre les méthodes

La pertinence des assignations produites par notre programme est vérifiée par trois critères : le taux d'accord global entre différentes méthodes (score C_3), le taux de recouvrement des segments (score SOV) et la distribution de longueur des segments.

Scores C_3

Les tableaux 2.5, 2.6, 2.7 et 2.8 rapportent les scores C_3 obtenus lors des comparaisons des paires d'assignations sur les structures des jeux de données *HRes set*, *MRes set*, *LRes set* et *RMN set*.

TAB. 2.5 – Scores C_3 entre paires de méthodes calculés sur le *HRes set*

	DSSP	STRIDE	PSEA	SECSTR	XTLSSTR	PDB
KAKSI	82.1%	83.5%	81.5%	81.7%	78.3%	83.4%
DSSP		95.4%	80.1%	93.4%	80.4%	90.8%
STRIDE			81.1%	91.9%	80.8%	89.9%
PSEA				79.8%	75.8%	78.1%
SECSTR					79.6%	87.4%
XTLSSTR						80.7%

TAB. 2.6 – Scores C_3 entre paires de méthodes calculés sur le *MRes set*

	DSSP	STRIDE	PSEA	SECSTR	XTLSSTR	PDB
KAKSI	82.7%	83.9%	82.0%	82.2%	78.5%	84.0%
DSSP		94.9%	80.8%	93.4%	80.5%	91.5%
STRIDE			81.7%	92.0%	80.7%	90.4%
PSEA				80.5%	76.3%	79.1%
SECSTR					79.8%	88.1%
XTLSSTR						80.7%

TAB. 2.7 – Scores C_3 entre paires de méthodes calculés sur le *LRes set*

	DSSP	STRIDE	PSEA	SECSTR	XTLSSTR	PDB
KAKSI	82.6%	83.7%	80.5%	82.5%	77.3%	83.5%
DSSP		93.4%	79.3%	93.1%	77.5%	91.5%
STRIDE			80.1%	91.0%	76.9%	89.3%
PSEA				78.8%	73.4%	77.9%
SECSTR					76.6%	88.3%
XTLSSTR						77.3%

TAB. 2.8 – Scores C_3 entre paires de méthodes calculés sur le *NMR set*

	DSSP	STRIDE	PSEA	SECSTR	XTLSSTR	PDB
KAKSI	84.7%	85.5%	80.1%	84.9%	78.7%	85.2%
DSSP		94.3%	81.0%	94.4%	79.9%	92.0%
STRIDE			81.6%	92.3%	79.4%	90.3%
PSEA				80.7%	74.5%	79.1%
SECSTR					78.9%	89.3%
XTLSSTR						78.5%

Un groupe de méthode a des scores C_3 particulièrement élevés entre paires, il s'agit de DSSP, STRIDE, SECSTR et PDB dont les scores C_3 vont de 87.4% (SECSTR versus PDB) à 95.4% (STRIDE versus DSSP). La forte ressemblance entre les assignations produites par DSSP et celles produites par STRIDE, deux méthodes qui utilisent la détection des liaisons hydrogènes, a déjà été noté ailleurs [115, 55, 68]. L'algorithme de SECSTR dérive de DSSP, il est donc logique de le retrouver dans ce groupe. Les descriptions de la PDB sont très similaires aux assignations de DSSP, ce qui est attendu d'après la façon dont sont générés ces champs.

Les assignation générées par XTLSSTR sont les plus divergentes, avec des scores C_3 tous inférieurs à 81%.

KAKSI and PSEA forment un groupe intermédiaire avec des scores C_3 compris entre 81.5% (KAKSI/PSEA) et 83.5% (KAKSI/STRIDE).

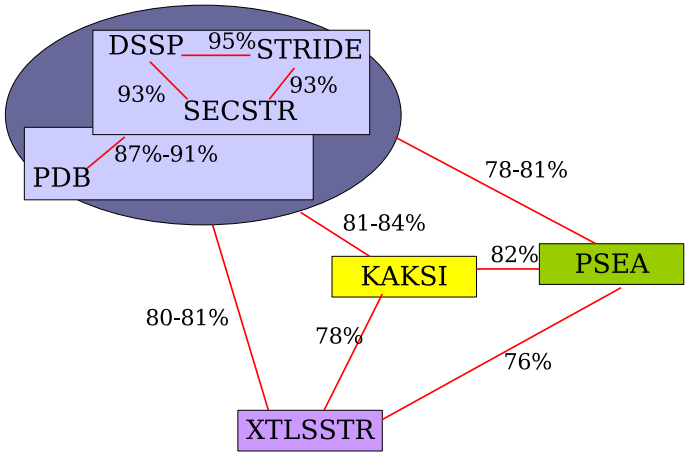

FIG. 2.10 – Aperçu graphique des scores C_3 obtenus sur le *HRes set*

Cette analyse fait apparaître les assignations produites par KAKSI comme une sorte de consensus, étant à des distances comparables (et relativement faibles) des autres méthodes, comme le montre le schéma de la figure 2.10.

Recouvrement des segments

Le score SOV est habituellement employé pour évaluer des méthodes de prédiction de structure secondaire. Ici, nous effectuons une comparaison entre deux assignations alternatives. Le calcul du SOV nécessitant de choisir une assignation *référence*, nous avons choisi de fixer les assignations de KAKSI comme référence pour permettre la comparaison. Le tableau 2.9 rapporte les scores SOV calculés pour les hélices et les feuillets lors de la comparaison entre KAKSI et les autres méthodes.

Nous discutons ici les scores obtenus sur le *HRes set*. En ce qui concerne les hélices, le plus haut score est obtenu lors de la comparaison entre KAKSI et DSSP (91.7%). Un score comparable est obtenu avec STRIDE. Avec les autres méthodes, les scores sont moins élevés mais restent supérieurs à 87%. Le recouvrement des brins assignés par KAKSI est assez bon avec DSSP, STRIDE et PDB, avec des SOV voisins de 90%. Des scores plus faibles sont

TAB. 2.9 – Scores SOV entre KAKSI et les autres méthodes, calculés sur tous les jeux de données avec KAKSI comme référence

Données	*HRes set*		*MRes set*		*LRes set*		*NMR set*	
Méthode	SOV_H [a]	SOV_b [b]	SOV_H	SOV_b	SOV_H	SOV_b	SOV_H	SOV_b
DSSP	91.7%	92.1%	91.4%	92.4%	90.6%	88.6%	91.6%	89.1%
STRIDE	91.2%	91.9%	89.9%	91.4%	89.9%	90.0%	92.1%	91.2%
SECSTR	89.0%	83.9%	87.5%	83.2%	88.4%	84.2%	90.8%	85.0%
PSEA	87.5%	82.7%	87.9%	84.3%	88.3%	82.6%	89.6%	82.3%
XTLSSTR	89.3%	73.4%	87.3%	73.2%	86.2%	63.5%	88.9%	63.5%
PDB	88.4%	89.4%	89.1%	89.8%	89.3%	86.4%	91.0%	91.1%

[a]SOV des hélices
[b]SOV des brins

obtenus avec les assignations de PSEA et SECSTRC. L'accord avec XTLSSTR est modéré (73.4% seulement). Le score C_3 entre XTLSSTR et KAKSI n'était que de 78.3% (voire table 2.5).

Par la suite, certaines comparaisons sont restreintes à KAKSI, STRIDE et PSEA sur les structures du *HRes set*. STRIDE est choisi car cette méthode est très employée. Les assignations produites par STRIDE sont très proches de celles de DSSP, la méthode la plus populaire. De plus c'est STRIDE qui donne les résultats les plus proches de KAKSI. La méthode PSEA est retenue pour la comparaison détaillée, car elle utilise une définition des structures secondaires relativement originale comparée aux méthodes basées sur les liaisons hydrogènes et à KAKSI, et donne des scores SOV raisonnables comparé à KAKSI.

Distributions des longueurs de segments

Les distributions de longueurs des hélices et des brins décrits par KAKSI, STRIDE et PSEA dans les structures du *HRes set* sont représentées sur la figure 2.11.

Trois zones peuvent être distinguées dans la distribution des longueurs d'hélices. (i) Pour les hélices de moins de 8 résidus, les distributions diffèrent sensiblement : STRIDE assigne beaucoup d'hélices de 3 résidus (1238 segments), alors que PSEA, et KAKSI n'assignent pas d'hélices de moins de 5 résidus. La distribution des hélices assignées par KAKSI montre un très fort pic à 7 résidus. (ii) Pour les hélices de 8 à 15 résidus, de faibles différences sont observées : la distribution générée par KAKSI montre un pic aux alentours de 12

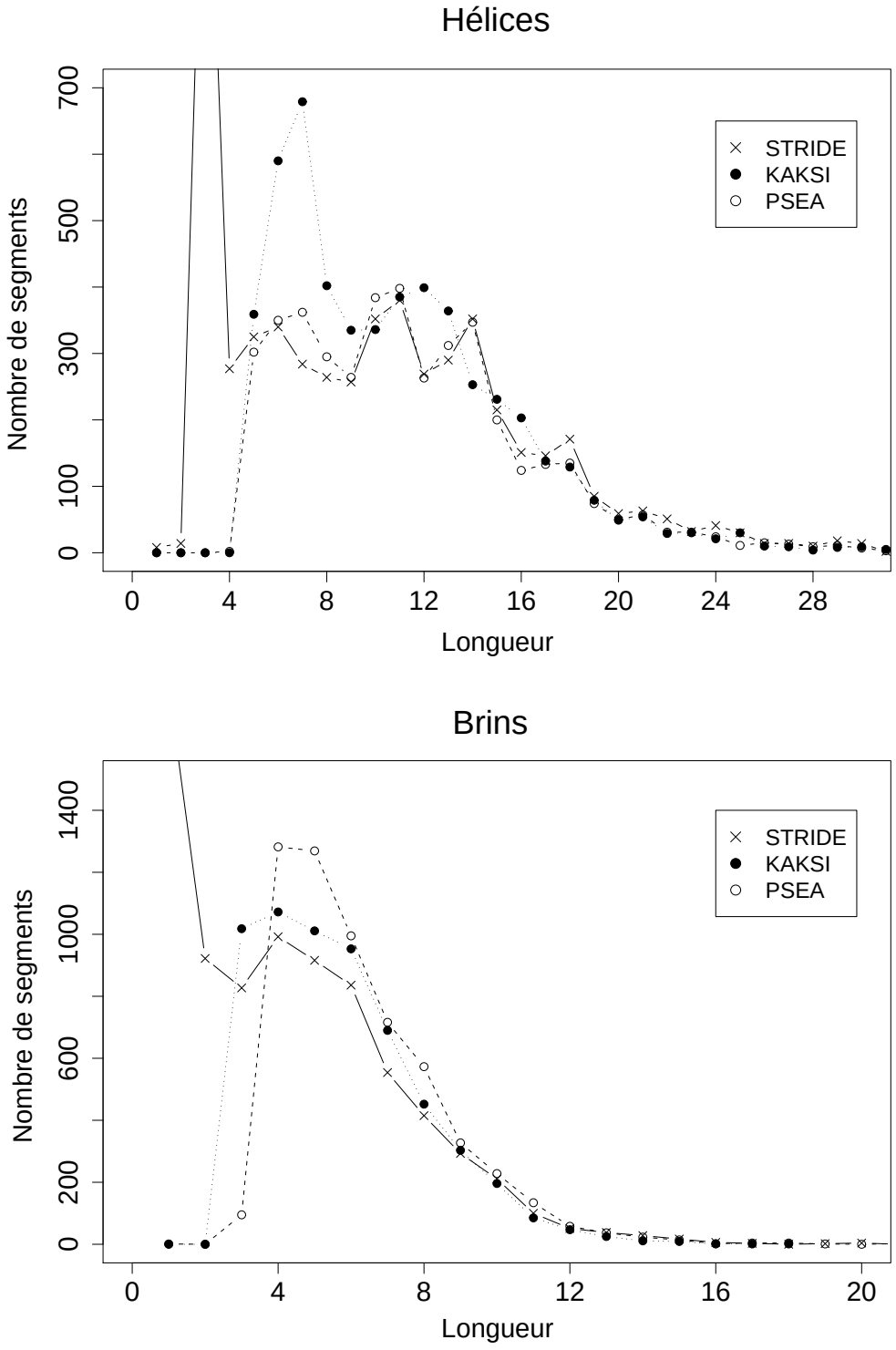

FIG. 2.11 – Distribution des longueurs d'hélices (en haut) et de brins (en bas) obtenues avec STRIDE, PSEA et KAKSI, sur les structures du *HRes set*.

résidus, contrairement aux distributions de PSEA et STRIDE. (iii) Au-delà de 15 résidus, les distributions sont similaires.

De la même manière, trois zones peuvent être distinguées dans les distributions de longueurs de brins. (i) Jusqu'à 6 résidus, les distributions de PSEA et KAKSI montrent des pics plus importants que STRIDE, aux alentours de 3 résidus pour KAKSI et 4 à 5 résidus pour PSEA. PSEA et KAKSI n'assignent pas de brins de moins de 3 résidus, contrairement à STRIDE qui assigne 1800 brins de 1 résidu qui correspondent à l'état b (*isolated beta bridge*). (ii) De 6 à 9 résidus, les brins assignés par PSEA et KAKSI sont plus nombreux que ceux assignés par STRIDE. (iii) Au-delà de 9 résidus, les distributions sont similaires.

Ces mesures globales (scores C_3 et SOV, distributions des longueurs) montrent que les assignations de KAKSI sont cohérentes avec celles des méthodes existantes. Dans la section suivante, nous essayons d'examiner plus en détail les différences entre ces trois méthodes, en particulier le traitement des extrémités des segments et des déformations dans les éléments de structures secondaires périodiques.

2.3.4 Comparaison détaillée

Longueurs des paires de segments assignés

Le score SOV permet de mesurer globalement le recouvrement entre segments produits par deux assignations différentes. Il ne permet pas de tirer d'information sur l'effet de la longueur des segments (par exemple, le recouvrement pourrait être meilleur pour des segments très longs) ou sur les longueurs des segments en correspondance (par exemple, une méthode pourrait assigner des segments systématiquement un peu plus longs qu'une autre). Une analyse plus fine consiste à examiner les longueurs respectives des paires de segments en correspondance lors de la comparaison de deux assignations. Ces paires sont celles utilisées pour le calcul du SOV : une paire est définie dès que deux assignations ont au moins un résidu dans la même structure secondaire. Les segments isolés (sans équivalent dans l'une des assignations) sont ignorés.

Si l'on considère les assignation de KAKSI comme la référence, 3 cas peuvent être distingués. Ils sont illustrés dans la figure 2.12. (i) Une *paire 1/1* correspond à un segment de KAKSI en correspondance avec un seul segment de l'autre méthode. (ii) Une *fusion* se produit lorsque KAKSI assigne un segment alors que l'autre méthode en définit plusieurs.

(iii) A l'inverse, une *division* correspond au cas où KAKSI définit plusieurs segments lorsque l'autre méthode en définit un seul.

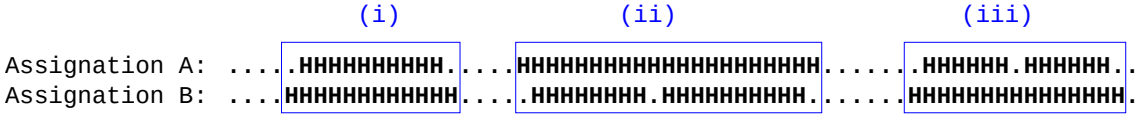

FIG. 2.12 – Trois configurations possibles pour les paires de segments de structure secondaire régulière selon deux assignations différentes. Avec l'assignation A prise comme référence, le cas (i) est une paire 1/1, (ii) est une fusion donnant lieu à deux paires et (iii) est une division qui génère deux paires également.

Longueur des hélices La figure 2.13 présente les bi-plots de longueurs d'hélices lors de la comparaison de KAKSI avec STRIDE et PSEA sur les structures de haute résolution. La comparaison de KAKSI et STRIDE génère 3755 *paires 1/1*, 1249 *divisions* et 142 *fusions*. Lors de la comparaison avec PSEA, nous observons 3443 *paires 1/1*, 1169 *divisions* et 150 *fusions*.

La forte accumulation des *paires 1/1* le long de la diagonale indique que KAKSI assigne beaucoup d'hélices de même longueur que STRIDE. Cette accumulation est plus forte en-dessous de la diagonale, ce qui indique que KAKSI assigne souvent des hélices un peu plus longues que STRIDE et PSEA. Cette répartition ne dépend pas de la longueur des hélices, mais elle est plus marquée pour la comparaison PSEA/KAKSI. La concentration des *paires 1/1* le long de la diagonale peut être mesurée par le coefficient de corrélation r^2 calculé sur les paires de longueurs. $r^2 = 0.88$ pour la comparaison KAKSI/STRIDE et $r^2 = 0.85$ pour la comparaison KAKSI/PSEA.

Les bi-plots des divisions montrent une forte accumulation de points correspondant à des hélices courtes (5 à 9 résidus) assignées par kaksi, correspondant à des hélices longues (10 à 20 résidus) assignées par STRIDE et PSEA. A l'inverse, les cas de fusions montrent des correspondances entre des hélices longues assignées par KAKSI avec des hélices courtes assignées par STRIDE et PSEA. Pour les deux comparaisons, les cas de *divisions* sont beaucoup plus fréquents que les cas de *fusions*. Ces définitions divergentes mettent en lumière une caractéristique de notre méthode d'assignation qui a tendance à décrire certaines longues hélices comme plusieurs petites. Cette caractéristique est une conséquence de la détection aboutie des coudes par KAKSI (voir analyse des géométries d'hélices). Cette tendance explique par ailleurs que KAKSI assigne un plus grand nombre

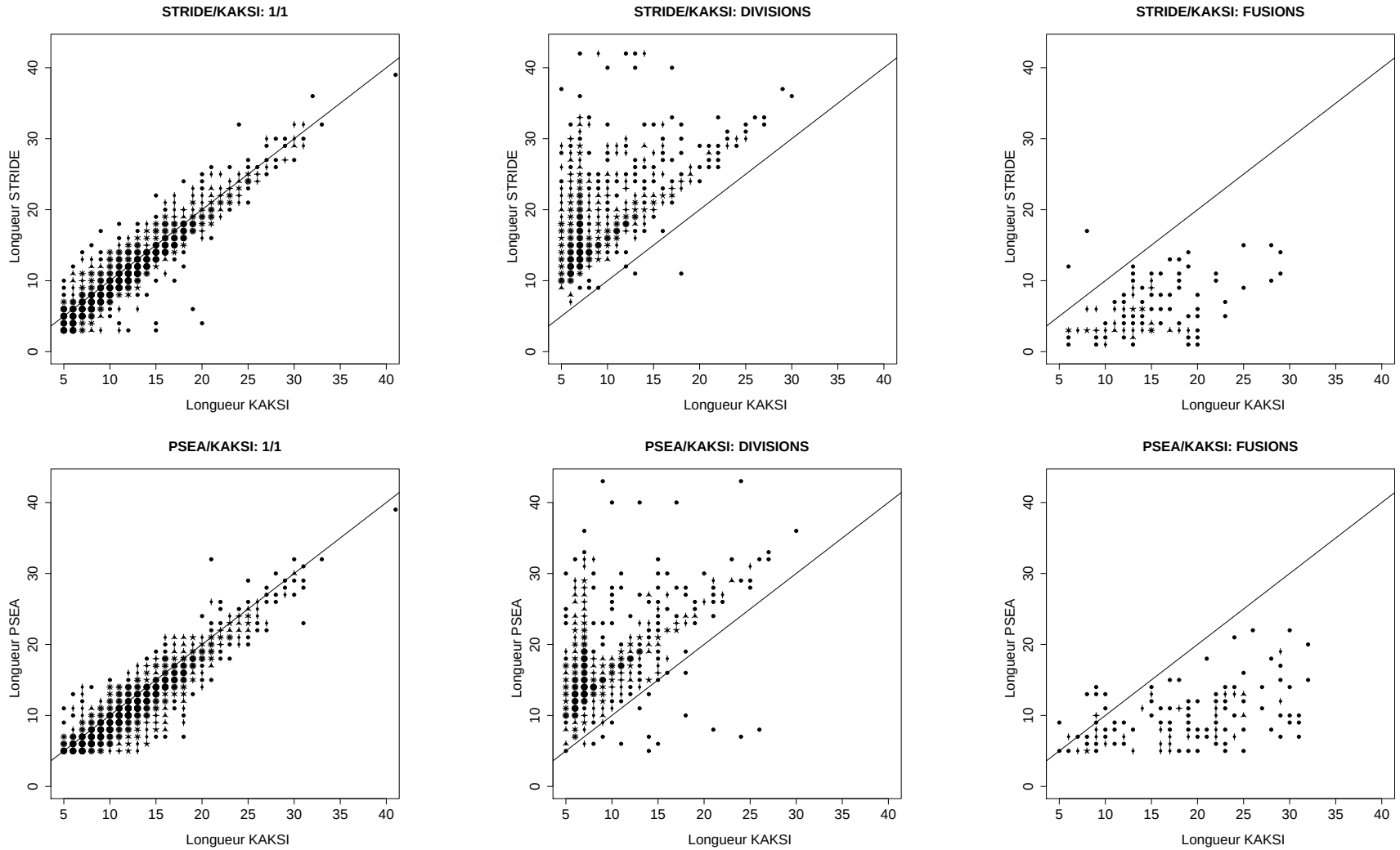

FIG. 2.13 – Paires de longueurs d'hélices issues de la comparaison des assignations de KAKSI et STRIDE (partie supérieure) et KAKSI et PSEA (partie inférieure) sur les structures du *HRes set*. Les trois types de paires sont portées sur des graphiques séparés. Les données sont reportées sous forme de *sunflower plot* : un point correspond à une observation, le nombre de *feuilles* est proportionnel au nombre d'observations supplémentaires. La diagonale x=y (longueurs de segments identiques) est indiquée.

d'hélices courtes que STRIDE et PSEA (voir distribution des longueurs).

Certains cas de *divisions* et de *fusions* apparaissent du mauvais côté de la diagonale. Il s'agit d'hélices coudées avec un désaccord sur la localisation du coude, comme illustré dans la figure 2.14.

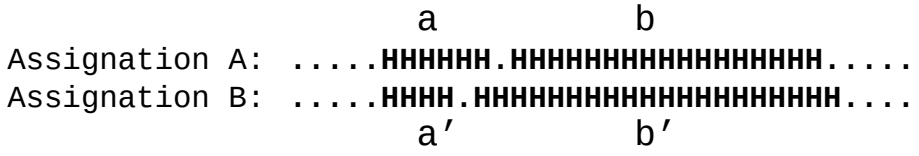

FIG. 2.14 – Cas de désaccord sur la localisation d'un coude dans une hélice. La paire a/b' est à la fois une *fusion* (à cause de la paire a/a') et une *division* (a cause de la paire b/b').

Longueurs de brins Les paires de longueurs de brins résultant des comparaisons de KAKSI avec STRIDE et PSEA sont rapportées sur la figure 2.15.

La comparaison de KAKSI avec STRIDE génère 5403 *paires 1/1* ($r^2 = 0.69$), 357 *divisions* et 214 *fusions*. La comparaison de KAKSI avec PSEA génère 4694 *paires 1/1* ($r^2 = 0.44$), 133 *divisions* et 214 *fusions*.

L'accord sur les longueurs est donc moins bon que pour les hélices et le phénomène des divisions est beaucoup moins marqué. 52 % des *paires 1/1* dans la comparaison KAKSI/STRIDE sont en-dessous de la diagonale et 22% sont au-dessus, ce qui indique que les brins sont majoritairement plus longs quand ils sont assignés par KAKSI. A l'inverse, pour la comparaison KAKSI/PSEA, 23% des *paires 1/1* sont en-dessous de la diagonale et 50% au-dessus. KAKSI assigne donc des brins plus courts que la méthode PSEA.

Pour compléter cette analyse, nous avons vérifié qu'il n'y a pas de décalage systématique entre deux assignations (par exemple, des hélices dont le début et la fin sont décalés en direction C-terminale). Pour cela, les valeurs des décalages aux extrémités N et C-terminales des hélices et des brins sont récoltées séparément. Les distributions obtenues sont identiques pour les décalages N et C-terminaux, ce qui indique que les désaccords de longueurs se font indifféremment aux deux extrémités, sans biais systématique.

Analyse de la géométrie des hélices avec HELANAL

En développant KAKSI, un soin particulier a été apporté à la détection des irrégularités dans les hélices. L'étude des longueurs de segments montre une tendance de KAKSI à

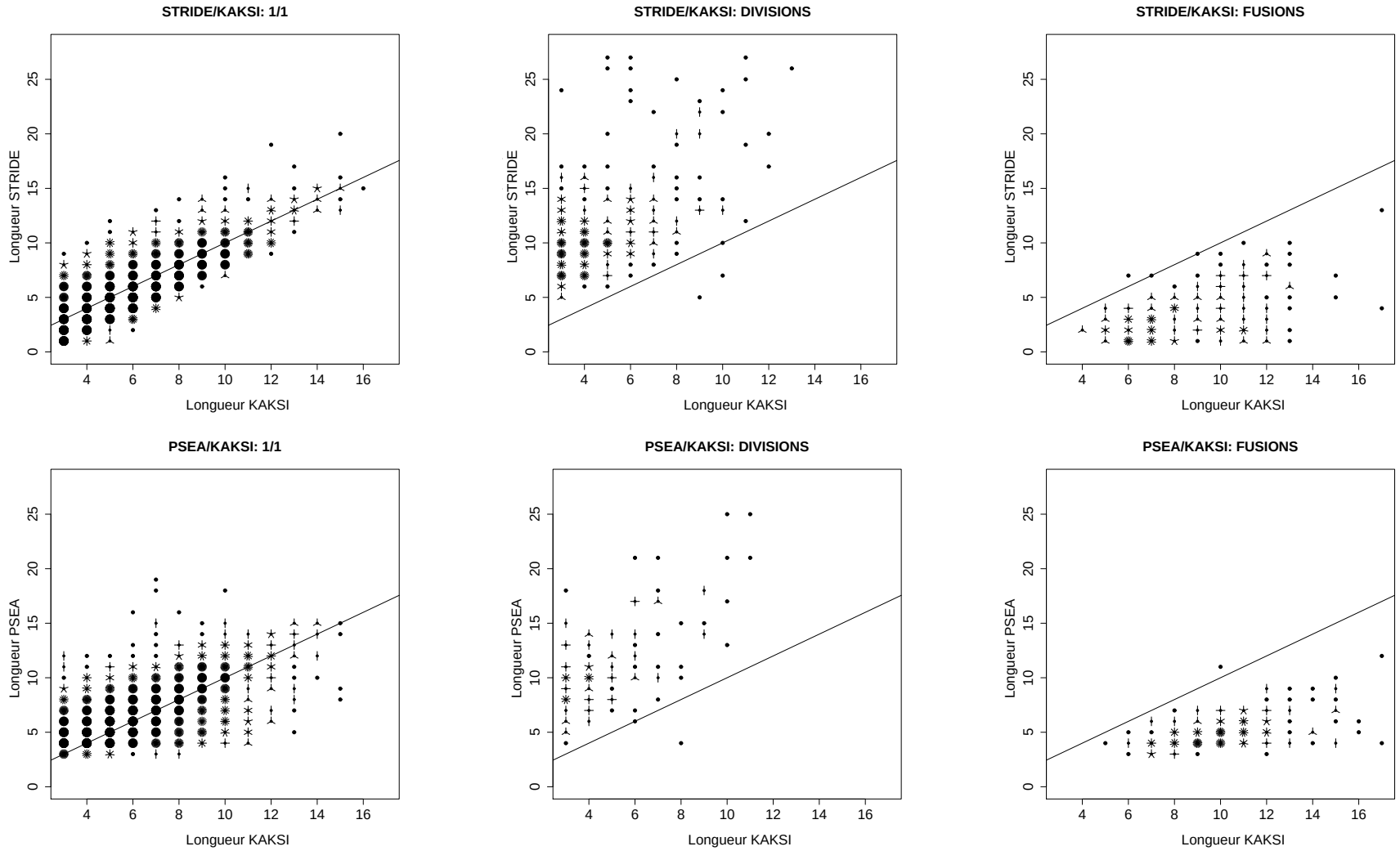

FIG. 2.15 – Paires de longueurs de brins issues de la comparaison des assignations de KAKSI et STRIDE (partie supérieure) et KAKSI et PSEA (partie inférieure) sur les structures du *HRes set*.

assigner certaines longues hélices sous la forme de plusieurs petites. L'analyse de la géométrie des hélices doit nous permettre d'évaluer l'efficacité de la détection des coudes sur la géométrie des hélices obtenues. Nous nous attachons en particulier à la répartition des hélices dans chacune des trois classes de géométrie définies par HELANAL : linéaire (L), courbée (C) et coudée (K). Les hélices non classées représentent moins de 1% des hélices sur nos données.

Dans un premier temps, nous classifions toutes les hélices de plus de 9 résidus, en utilisant les limites d'hélices définies par chacune des différentes méthodes d'assignation. Cette analyse montre qu'une forte proportion des hélices sont coudées. Sur les structures du *HRes set*, environ 20% des hélices assignées par DSSP, STRIDE et KAKSI sont coudées et jusqu'à 30% des hélices assignées par SECSTR et XTLSSTR. Cette proportion n'est que de 16% pour les assignations de la PDB, et moins de 10% pour les assignations de PSEA. Elle est plus élevée sur les structures de basse résolution. Sur les structures RMN, on obtient jusqu'à 40% d'hélices coudées dans les assignations de PSEA et plus de 50% pour STRIDE, SECSTR et PDB (48% pour KAKSI).

Ce fort taux d'hélices coudées est en accord avec les résultats obtenus par Barlow et collaborateurs [15]. Cependant, il est bien supérieur a celui rapporté par Kumar et Bansal, les auteurs de HELANAL : environ 4 % d'hélices coudées [114]. L'approche de Kumar et Bansal était différente de la notre : dans leur étude, les assignations des hélices par DSSP sont modifiées au cas par cas. En utilisant des critères de distance et d'angle, les hélices sont raccourcies pour éliminer les distorsions aux extrémités. La forte proportion d'hélices coudées que nous observons est sans doute due à ces résidus terminaux. Nous avons donc décidé d'appliquer une correction systématique en raccourcissant toutes les hélices d'un résidu à chaque extrémité. La raison de cette correction systématique (et non pas au cas par cas comme Kumar et Bansal) est que nous voulons faire une analyse statistique de la géométrie des hélices telles qu'elles sont décrites par les différents algorithmes. Une correction basée sur des critères géométriques masquerait probablement les différences existant entre les différentes définitions d'hélices. Dans les tableaux 2.10, 2.11, 2.12 et 2.13, nous rapportons les résultats obtenus avant et après correction.

HELANAL analyse les hélices à partir de 9 résidus. Notre étude est donc restreinte aux hélices d'au moins 11 résidus, raccourcies des deux côtés. Nous discutons ici les résultats obtenus sur le *HRes set*. La correction des hélices fait sensiblement baisser le taux

TAB. 2.10 – Géométrie des hélices analysées par le programme HELANAL sur les structures du HRes set.

Méthode	Sans correction				Avec Correction [a]			
Longueur minimale	11				9 après correction			
	%L[b]	%C[c]	%K[d]	N[e]	%L	%C	%K	N
DSSP	8.3	70.0	21.2	2215	10.9	70.8	17.8	2215
STRIDE	10.1	65.9	23.6	2431	10.8	68.5	20.2	2431
PSEA	10.9	78.5	10.0	2260	11.5	80.0	7.8	2260
SECSTR	8.0	55.7	36.0	2349	10.0	59.7	29.9	2349
XTLSSTR	8.7	58.9	32.1	2618	9.5	61.4	28.9	2618
KAKSI	10.2	66.5	22.8	2442	12.3	72.6	14.5	2442
PDB	11.4	71.1	17.0	2565	11.3	71.5	12.0	2565

[a]les assignations sont corrigées en retirant un résidu à chaque extrémité
[b]pourcentage d'hélices classifiées linéaires par HELANAL
[c]pourcentage d'hélices classifiées courbées par HELANAL
[d]pourcentage d'hélices classifiées coudées par HELANAL
[e]nombre d'hélices soumises à l'analyse

TAB. 2.11 – Géométrie des hélices analysées par le programme HELANAL sur les structures du *MRes set*

Méthode	Sans correction				Avec Correction			
Longueur minimale	11				9 après correction			
	%L	%C	%K	N	%L	%C	%K	N
DSSP	10.79	64.43	24.15	2381	12.47	66.74	20.20	2381
STRIDE	10.99	64.04	24.43	2567	12.66	65.52	21.08	2567
PSEA	12.18	73.22	13.81	2390	13.26	74.31	11.59	2390
SECSTR	9.10	54.93	35.57	2494	11.15	58.38	29.99	2494
XTLSSTR	9.81	60.30	29.52	2690	11.71	61.12	26.88	2690
KAKSI	11.26	64.02	23.95	2593	13.23	70.15	15.77	2593
PDB	12.23	65.98	21.03	2763	13.86	69.56	15.92	2763

d'hélices coudées, ce qui montre qu'une partie des coudes observés est effectivement due aux déformations aux extrémités. Après correction, le taux d'hélices coudées assignées par KAKSI (14.5%) est le plus proche de celui de la PDB (12% d'hélices coudées) parmi tous les algorithmes d'assignation testés. Les assignations générées par KAKSI sont par ailleurs celles qui définissent la plus grande proportion d'hélices linéaires. PSEA génère très peu d'hélices coudées (7.8% seulement) mais le nombre d'hélices analysées est un peu plus

TAB. 2.12 – Géométrie des hélices analysées par le programme HELANAL sur les structures du *LRes set*

Méthode	Sans correction				Avec Correction			
Longueur minimale	11				9 après correction			
	%L	%C	%K	N	%L	%C	%K	N
DSSP	15.04	54.60	29.57	1390	18.55	57.80	23.08	1391
STRIDE	14.06	50.65	34.47	1465	16.38	54.81	27.99	1465
PSEA	15.55	62.41	20.90	1402	19.40	63.41	15.98	1402
SECSTR	12.07	44.10	42.74	1467	15.80	48.84	34.67	1468
XTLSSTR	13.57	53.19	31.64	1378	17.34	53.41	27.65	1378
KAKSI	16.17	57.20	25.71	1416	20.20	61.09	17.58	1416
PDB	14.58	54.98	29.74	1577	17.44	59.80	21.81	1577

TAB. 2.13 – Géométrie des hélices analysées par le programme HELANAL sur les structures du *NMR set*

Méthode	Sans correction				Avec Correction			
Longueur minimale	11				9 après correction			
	%L	%C	%K	N	%L	%C	%K	N
DSSP	7.20	46.54	45.43	361	11.63	52.35	35.18	361
STRIDE	7.07	40.91	51.26	396	11.87	45.20	41.92	396
PSEA	7.99	51.24	39.94	363	13.22	53.99	31.40	363
SECSTR	6.53	37.86	54.83	383	8.36	46.48	44.39	383
XTLSSTR	6.15	47.33	45.72	374	7.75	54.81	35.56	374
KAKSI	7.32	43.94	47.98	396	13.38	50.00	35.86	396
PDB	9.15	39.44	50.70	426	11.74	49.30	38.50	426

faible que pour KAKSI.

Pour faire le lien avec l'étude des paires de longueurs, il est intéressant d'examiner la géométrie des hélices impliquées dans des *divisions* (définition de plusieurs hélices par KAKSI et une seule hélice par STRIDE). Ainsi, 128 paires de type *division* impliquent des hélices de plus de 9 résidus pour lesquelles l'hélice assignée par STRIDE est coudée tandis qu'une des hélices assignée par KAKSI est linéaire ou courbée. Certains cas sont représentés dans les figures 2.16 à 2.18. Le cas inverse (hélice KAKSI coudée *vs* hélice STRIDE linéaire ou courbée) ne concerne que 7 paires. La division de longues hélices en plusieurs petites permet donc de définir des hélices dépourvues de coudes.

Toutes ces observations suggèrent que la détection des coudes implémentée dans KAKSI

est efficace et génère des assignations des limites d'hélices très pertinentes. La caractéristique des assignations de KAKSI réside donc dans la géométrie des hélices α : tout en assignant des hélices souvent plus longues que STRIDE, la géométrie des hélices reste satisfaisante, avec même un plus fort taux d'hélices linéaires que les autres méthodes d'assignation, et un taux d'hélices coudées très proches des assignations de la PDB. Nous en montrons quelques exemples dans la section suivante.

2.4 Quelques exemples de structures analysées différemment par les programmes d'assignation

Nous rapportons, dans les figures 2.16, 2.17, 2.18 et 2.19 quelques exemples de structures pour lesquelles les assignations de STRIDE et KAKSI sont qualitativement très différentes.

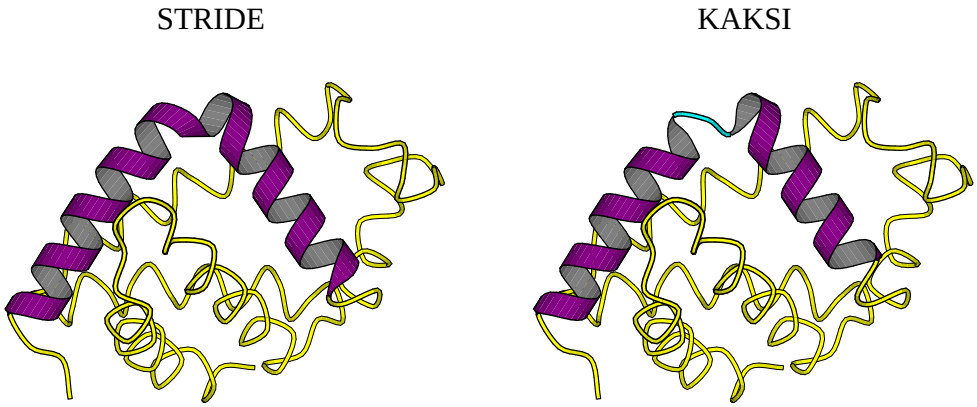

FIG. 2.16 – Désaccord dans l'assignation de la région 4-35 de l'hémoglobine I de clam Lucina pectinata (code PDB : 1bob, résolution : 1.43 Å).

Les trois premiers exemples concernent l'assignation de longues hélices. Dans la structure PDB 1bob (figure 2.16), STRIDE assigne une hélice α du résidu 4 au résidu 35. Cette hélice montre un coude très marqué au résidu 20. Dans cette région KAKSI assigne 2 hélices distinctes du résidu 4 au résidu 19 et du résidu 21 à 34. Les courbures moyennes de ces hélices, calculées par HELANAL sont respectivement de 3.84° et 9.0°. L'hélice 4-19 est analysée comme courbée par HELANAL. L'hélice 21-34 est classifiée coudée, mais elle devient linéaire après suppression des résidus terminaux. Ces deux hélices forment un angle de 83°.

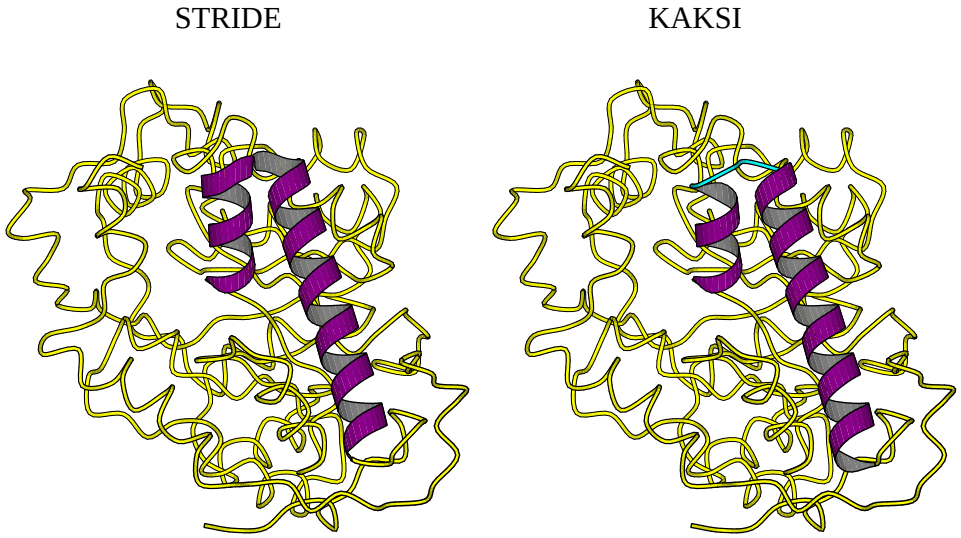

FIG. 2.17 – Désaccord dans l'assignation de la région 308-341 de la chaîne A de la L(+)-mandelate déshydrogenase de Pseudomonas *putida* (code PDB : 1p4c, résolution : 1.35 Å).

Le deuxième exemple (figure 2.17) est encore plus frappant. Il s'agit de la structure PDB 1p4c. STRIDE assigne une hélice α de 33 résidus, du résidu 308 au résidu 340, alors que dans cette région, la chaîne principale revient sur elle-même. L'assignation de KAKSI consiste en deux hélices du résidu 308 au résidu 315 (cette hélice est trop courte pour être analysée par HELANAL) et du résidu 320 au résidu 341 (courbure moyenne 4.3°, analysée comme linéaire par HELANAL).

Le troisième exemple illustre le cas d'une très longue hélice divisée en trois morceaux, dans la région 21-62 de la structure 1jb0, chaîne B. Bien que moins marqués que dans les deux premiers exemples, les coudes dans l'hélice sont bien visibles. STRIDE assigne une hélice du résidu 21 au résidu 62 (courbure moyenne 13.1°) alors que KAKSI assigne trois hélices : résidus 21 à 33 (courbure moyenne 4.5°), 35 à 46 (courbure moyenne 3.0°) et 48 à 61 (courbure moyenne 6.6°). Ces trois hélices sont analysées comme courbées.

Le dernier exemple concerne l'assignation divergente de feuillets β dans la région 61-136 de la structure PDB 1OD3, chaîne A. STRIDE assigne deux très longs brins β des résidus 61 à 82 et 116 à 135, qui permettent à la chaîne principale de revenir sur elle-même. Chacun de ces longs brins est divisé en deux parties par KAKSI : résidus 61 à 69, 75 à 83, 115 à 122 et 128 à 136. Bien que KAKSI n'intègre pas de critère spécifique pour détecter les déformations dans les brins, les critères d'assignation relativement sévères peuvent aboutir à la division de certains brins.

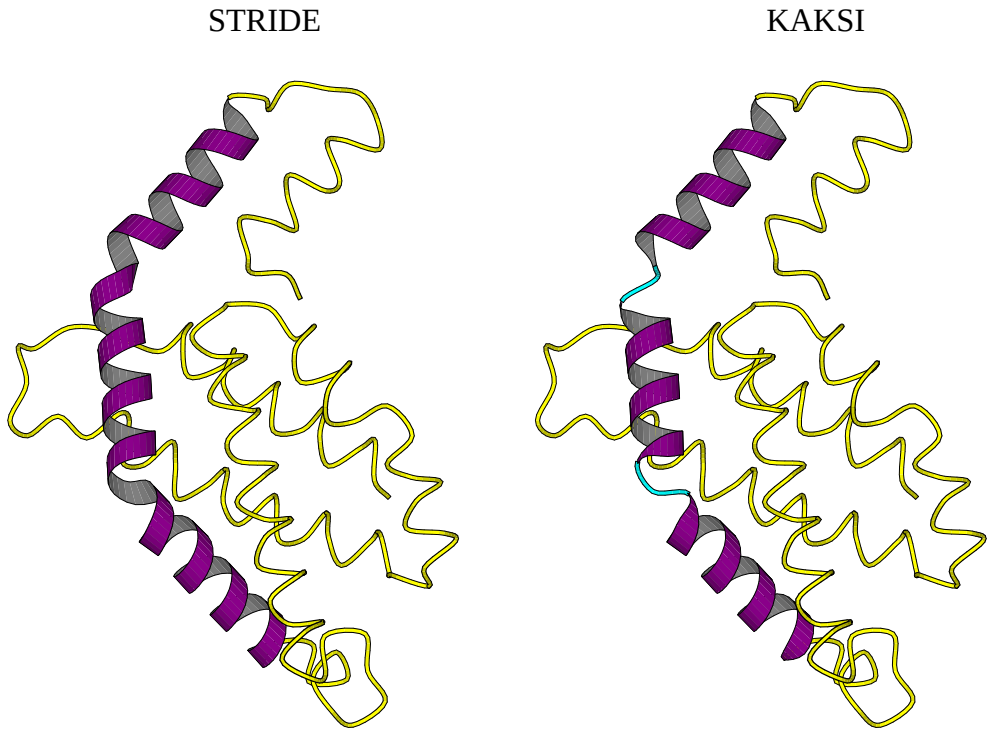

FIG. 2.18 – Désaccord dans l'assignation de la région 21-62 de la chaîne B de la C-phycocyanine de thermophylic cyanobacterium Synechococcus *elongatus* (code PDB : 1jb0, résolution : 1.45 Å).

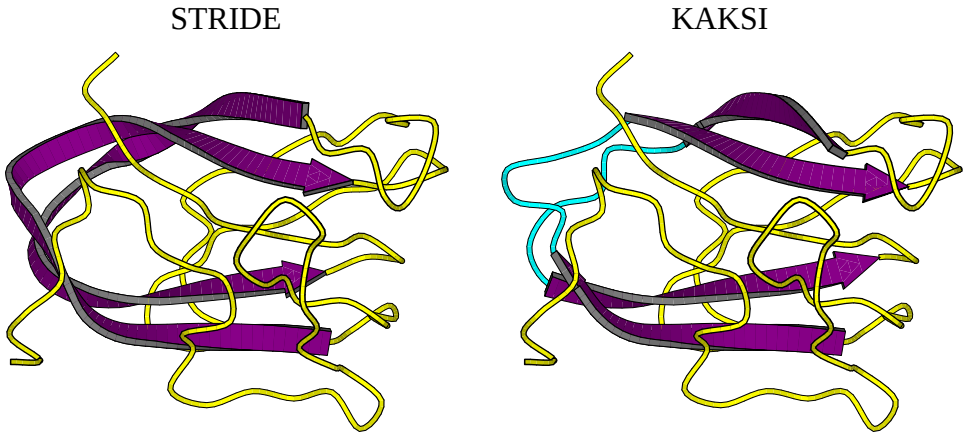

FIG. 2.19 – Désaccord dans l'assignation de la région 61-136 de la chaîne A de l'endo-xylanase de Clostridium *stercorarium* (code PDB : 1OD3, résolution : 1 Å).

Ces exemples visuels sont d'autant plus marquants que les différences entre les deux assignations ne porte que sur quelques résidus, mais change radicalement la description qualitative de la structure 3D. Le programme KAKSI permet donc de fournir une description plus pertinente des structures 3D.

2.5 Conclusion

Nous avons développé un nouvel outil d'assignation des structures secondaires d'après les structures 3D. Notre méthode, appelée KAKSI, utilise les distances entre Cα et les angles dièdres Φ/Ψ. Ces critères ont déjà été utilisés par ailleurs par d'autres algorithmes, mais notre méthode comporte en plus des critères spécifiques pour affiner la détection des coudes dans les hélices. Comme la plupart des programmes d'assignation (à l'exception de PSEA), notre méthode est sensible à la résolution et au type de technique utilisée pour la détermination expérimentale. En conséquence, nous suggérons d'adapter les paramètres de la détection à la résolution, à la technique expérimentale et au type de structures secondaires, les feuillets β étant plus difficiles à assigner. La comparaison détaillée des assignations de KAKSI avec celles de cinq autres programmes ainsi que les descriptions de la PDB nous a permis de valider notre méthode et d'analyser les apports de KAKSI pour l'analyse des structures 3D. L'accord général avec les autres méthodes est tout à fait satisfaisant. Les hélices et les brins sont principalement de même longueur ou légèrement plus longs que les assignations de STRIDE. En cas de désaccord sur le nombre d'hélice assignées, les cas de *divisions* d'hélices en plusieurs petites sont plus nombreux que les cas de *fusions*. Des *divisions* de brins β sont aussi observée, mais de façon beaucoup moins marquée. L'étude de la géométrie des hélices à l'aide d'un programme externe montre que les hélices assignées par KAKSI sont plus régulières que celle assignées par les autres méthodes, à l'exception notable de PSEA. KAKSI est aussi la méthode qui génère des hélices avec une géométrie très proche des descriptions PDB, et le plus d'hélices linéaires. Comme le font remarquer Andersen et collaborateurs [5], chaque méthode reflète sa propre définition des structures secondaires. Notre définition favorise une certaine régularité des hélices et des brins, comme le montrent les figures 2.16, 2.17, 2.18 et 2.19.

Dans la suite de ce mémoire, les structures secondaires seront donc définies par KAKSI. L'objectif principal de cette thèse a été la mise au point d'une méthode de prédiction de structure locale des protéines à l'aide des modèles de chaînes de Markov cachés (HMM). La chapitre suivant présente ces modèles mathématiques et les stratégies existantes pour prédire la structure des protéines en utilisant les HMM.

Chapitre 3

Les HMM et leur utilisation en modélisation mathématique des protéines

Au cours de ma thèse, j'ai mis en oeuvre des modèles probabilistes appelés modèles de chaînes de Markov cachées (HMM : hidden Markov models[1]) afin de prédire la structure locale des protéines (structure secondaire et zones d'angles Φ/Ψ) d'après leurs séquences. Ce chapitre présente les HMM dans la section 3.1 et un état de l'art de l'utilisation des HMM pour la prédiction de structure locale des protéines dans la section 3.2.

3.1 Présentation des modèles HMM

Les HMM sont des modèles probabilistes permettant de modéliser des signaux hétérogènes, d'utilisation courante dans des domaines comme la reconnaissance de la parole [149]. Leur application à la modélisation des séquences biologiques remonte aux travaux de Churchill en 1989 [43]. Une introduction aux HMM pourra être trouvée dans [149] et en particulier pour la modélisation des séquences biologiques dans [56].

La modélisation HMM d'une séquence est caractérisée par un double processus :

- un processus caché modélisant la succession de plages homogènes de différentes propriétés,
- le processus observé : les effets visibles de ces différentes plages sur la séquence elle-même.

Le premier processus est dit caché car dans la pratique il n'est pas observable et ne manifeste qu'à travers ses effets sur la séquence.

Les HMM peuvent être utilisés en mode *génératif*, pour simuler des séquences. Pour cela, il faut commencer par simuler le processus caché, puis le processus observé, d'après le processus caché. Dans ce cadre, il est intuitif de dire que la séquence observée est *émise* par le processus caché. Pour l'analyse de séquences, les HMM sont utilisés en mode *analytique* : on suppose que la séquence a été simulée par le modèle considéré. Les propriétés de ce modèle permettent alors de faire des prédictions sur la séquence.

Le succès des HMM est dû à la très grande souplesse de modélisation offerte par ces modèles (hétérogénéité, modèles phasés, structure des données). De plus, il existe des algorithmes permettant de retrouver le chemin caché à partir de la séquence dont la complexité croît linéairement avec la taille de la séquence.

La section 3.1.1 a pour but de présenter la spécification du modèle : dépendances entre

[1]Le sigle HMM sera utilisé dans la suite de ce manuscrit.

les variables aléatoires qui le définissent, lois suivies par ces variables et paramètres qui y sont associés. La section 3.1.2 présente les algorithmes mis en œuvre pour la reconstruction du chemin caché quand les paramètres du HMM sont connus. Enfin, la section 3.1.3 présente les procédures d'estimation des paramètres dans différents cas de figure.

3.1.1 Spécification du modèle

Modèle de Markov

Dans une modélisation markovienne d'ordre l d'une séquence de lettres $X = X_1X_2X_3....X_n$, l'apparition de la lettre à la position t ne dépend que des l lettres précédentes. En terme de probabilités :

$$P(X_t \mid X_1X_2X_3.....X_{t-1}) = P(X_t \mid X_{t-l}X_{t-l+1}...X_{t-1})$$

Lorsque cette probabilité est constante pour toute valeur de t, la chaîne de Markov est dite homogène.

Si $l = 0$, il y a indépendance entre les sites. Si l est différent de zéro, le passé proche de la séquence influence le présent. Cette influence est représentée par les probabilités conditionnelles d'observer une lettre sachant les lettres précédentes. L'ensemble de ces probabilités forme la matrice de transition markovienne de la séquence. Par exemple, si la séquence prend ses valeurs dans un alphabet à 3 lettres *abc*, la matrice de Markov d'ordre 1 s'écrit :

$$\begin{pmatrix} P(X_t = a \mid X_{t-1} = a) & P(X_t = b \mid X_{t-1} = a) & P(X_t = c \mid X_{t-1} = a) \\ P(X_t = a \mid X_{t-1} = b) & P(X_t = b \mid X_{t-1} = b) & P(X_t = c \mid X_{t-1} = b) \\ P(X_t = a \mid X_{t-1} = c) & P(X_t = b \mid X_{t-1} = c) & P(X_t = c \mid X_{t-1} = c) \end{pmatrix}$$

Ces probabilités vérifient :

$$\forall y, \sum_x P(x \mid y) = 1.$$

Ainsi, $P(X_t = c \mid X_{t-1} = a)$ peut se déduire de $P(X_t = a \mid X_{t-1} = a)$ et $P(X_t = b \mid X_{t-1} = a)$. Le nombre de paramètres linéairement indépendants de la matrice de Markov est donc $2 \times 3 = 6$.

Une loi initiale π est nécessaire pour le choix de la première lettre :

$$\begin{pmatrix} P(X_1 = a) \\ P(X_1 = b) \\ P(X_1 = c) \end{pmatrix}$$

Cette loi pourra éventuellement être la loi stationnaire de la chaîne de Markov, ce qui ne rajoute pas de paramètres supplémentaires. La loi stationnaire est la loi de probabilité atteinte par la chaîne, sous certaines hypothèses, après un temps t infini (voir Annexe).

Dans le cas des séquences de protéines, la matrice de transition d'ordre 1 est une matrice 20×20, puisqu'il y a 20 acides aminés différents. Le nombre de paramètres linéairement indépendants de cette matrice est de 19×20.

HMM

La modélisation d'une séquence par une chaîne de Markov homogène suppose que la séquence a les mêmes propriétés statistiques sur toute sa longueur. Au contraire, la modélisation HMM permet de représenter des séquences *homogènes par plages*. Une séquence est alors vue comme un assemblage de morceaux de natures - on pourrait dire de textures - différentes, caractérisés par des matrices markoviennes différentes.

Dans le cas qui nous intéresse, la modélisation de la structure secondaire des protéines, il est tout naturel d'adopter ce type de modèle, puisque la composition des séquences en acides aminés est corrélée avec la structure secondaire. Différentes matrices de transitions markoviennes seront spécifiées pour les différentes structures secondaires.

Une modélisation HMM suppose que l'on se munisse de plusieurs matrices markoviennes, caractérisant les propriétés de différentes plages de la séquence. Ces plages peuvent être envisagées comme des *états cachés* visités par la séquence. Le processus de changement d'état le long de la séquence est lui-même géré par une chaîne de Markov. On obtient alors un modèle emboîté :

- le processus des changements d'états cachés qui régit l'apparition des différentes plages,
- la séquence qui est générée conditionnellement au premier processus.

Le processus de changement d'état est dit *caché* car il n'est observable qu'à travers ces effets sur la séquence modélisée. Par opposition, la séquence est appelée *processus observé*.

Les transitions entre états cachés peuvent être représentées par un graphe d'états. La figure 3.1 présente l'exemple d'un HMM à 3 états cachés : un état caché rouge, un état caché jaune, un état caché vert. Certaines transitions entre états cachés peuvent ne pas être autorisées. Dans cet exemple, une plage rouge ne peut pas être voisine d'une plage jaune.

Si nous reprenons l'exemple d'une séquence observée prenant ses valeurs dans un alphabet à 3 lettres *abc*, chacun des états cachés est caractérisé par une matrice de transition markovienne sur ces 3 lettres.

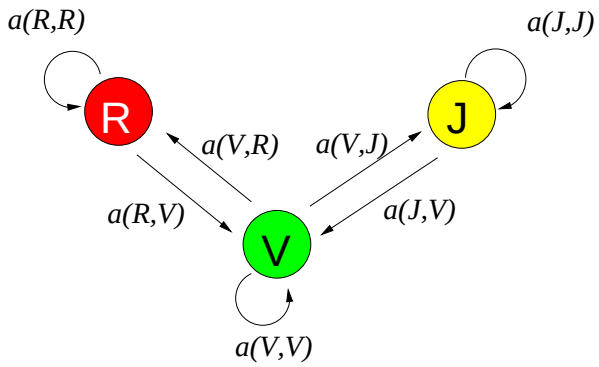

FIG. 3.1 – Graphe des états cachés d'un processus à 3 états : rouge(R), vert(V) et jaune(J). Les probabilités associées aux transitions sont portées sur le graphe. Seules les transitions de probabilités non nulles sont indiquées.

Notations

Les notations majuscules correspondent aux variables aléatoires, les notations minuscules aux valeurs prises par ces variables.

Le **processus caché** est noté S. Il prend ses valeurs dans l'alphabet $\mathcal{S}$ à nombre fini d'états cachés, m. L'état caché visité à la position t est noté S_t et la séquence partielle des états cachés $S_1 S_2 ... S_l$ est notée S_1^l. Le processus de changement d'états suit une chaîne de Markov d'ordre 1 de paramètres :

$a(u, v) = P(S_t = v \mid S_{t-1} = u)$, $\sum_v a(u, v) = 1$,

et de loi initiale : $\pi(u) = P(S_1 = u)$, $\sum_u \pi(u) = 1$.

La probabilité d'une suite d'états cachés $s_1 s_2 ... s_n$ s'écrit alors

$$P(S_1^n = s_1^n) = \pi(s_1) \prod_{t=1}^{n-1} a(s_t, s_{t+1}).$$

Le **processus observé** est noté X. Il prend ses valeurs dans un alphabet $\mathcal{X}$ de cardinal 20 dans le cas des protéines. La lettre observée à la position t est notée X_t. La séquence observée partielle $X_1 X_2 ... X_l$ est notée X_1^l.

Les paramètres de la chaîne de Markov des observations de l'état caché u sont :

$b_u(x) = P(X_t = x \mid S_t = u)$, $\sum_x b_u(x) = 1$.

$b_u(x) = P(X_t = x \mid S_t = u)$ est la probabilité d'émettre x dans l'état caché u. Ceci correspond à une chaîne de Markov d'ordre zéro sur les observations conditionnellement à l'état caché. Il est possible d'augmenter la mémoire du processus observé conditionnellement à l'état caché en considérant une dépendance markovienne d'ordre l. Les paramètres b s'écrivent alors $b_u(x; x_{t-l}^{t-1}) = P(X_t = x \mid S_t = u, X_{t-l}^{t-1} = x_{t-l}^{t-1})$. L'ordre l de la chaîne de Markov, peut éventuellement être différent pour chaque état caché. Pour la présentation des algorithmes de prédiction, on supposera qu'il y a indépendance entre les sites ($l = 0$). L'augmentation de l'ordre nécessite le remplacement des $b_u(x)$ par les $b_u(x; x_{t-l}^{t-1})$ appropriés.

La probabilité d'observer conjointement la suite d'états cachés $s_1 s_2 ... s_n$ et la séquence observée $x_1 x_2 .. x_n$ s'écrit

$$\begin{aligned} P(S_1^n = s_1^n, X_1^n = x_1^n) &= P(S_1^n = s_1^n) P(X_1^n = x_1^n \mid S_1^n = s_1^n) \\ &= \pi(s_1) \prod_{t=1}^{n-1} a(s_t, s_{t+1}) \prod_{t=1}^{n} b_{s_t}(x_t). \end{aligned}$$

La probabilité d'une séquence particulière $x_1 x_2 .. x_n$ est obtenue par

$$P(X_1^n = x_1^n) = \sum_{s_1^n} P(S_1^n = s_1^n) P(X_1^n = x_1^n \mid S_1^n = s_1^n).$$

En effet, une séquence donnée peut avoir été générée par différentes suites d'états cachés. Il faut donc tenir compte de toutes les suites d'états cachés possibles.

Un HMM est entièrement défini par la loi initiale des états cachés, les probabilités a - matrice de Markov des changements d'états - et b - matrices de Markov d'émission des observations, conditionnellement aux états cachés.

Les HMM sont des modèles graphiques. Il est possible de représenter les dépendances entre variables dans un DAG (*Directed Acyclic Graph*). La figure 3.2 présente le DAG

d'un modèle M1M0 : le processus caché est une chaîne de Markov d'ordre 1 et le processus observé est une chaîne de Markov d'ordre 0. La figure 3.3 présente le DAG d'un modèle M1M1.

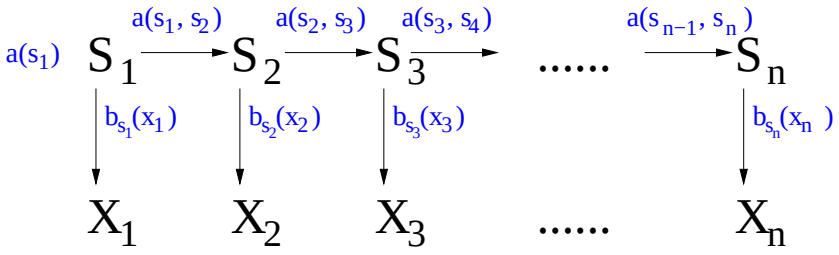

FIG. 3.2 – HMM M1M0.

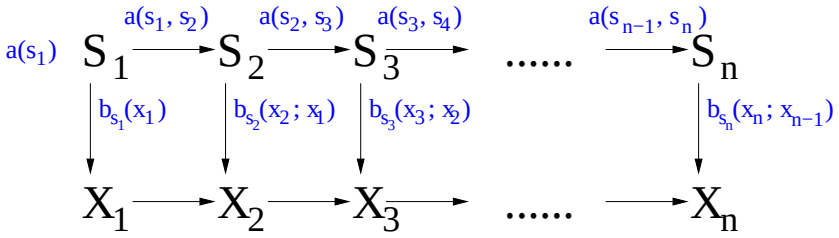

FIG. 3.3 – HMM M1M1.

3.1.2 Algorithmes de prédiction

Supposons, dans un premier temps, que nous disposons d'un HMM dont les paramètres sont connus. Dans une démarche prédictive, on aura modélisé les caractéristiques qui nous intéressent dans le processus caché. Pour la prédiction de la structure secondaire des protéines, la structure secondaire constitue le processus caché et la séquence est la séquence observée. La prédiction consiste alors à retrouver le processus caché d'après la séquence observée.

Deux algorithmes permettent de répondre à cette question, en adoptant deux points de vue différents :

– ce qui importe est de retrouver le chemin le plus probablement utilisé dans le graphe des états cachés pour générer la séquence observée. La réponse à cette attente est fournie par l'algorithme de *Viterbi*.

– ce qui importe est de trouver, pour chaque site de la séquence, l'état caché qui a le plus probablement généré la séquence observée, en tenant compte de tous les chemins possibles. Dans ce cas, l'algorithme approprié est l'algorithme *forward/backward*.

Ces deux algorithmes sont décrits notamment dans [56].

Algorithme de Viterbi

L'algorithme de Viterbi consiste à rechercher s^*, le chemin le plus probable dans le graphe des états cachés d'après la séquence observée :

$$s^* = argmax_s P(S \mid X).$$

Ceci revient a rechercher $argmax_s P(S, X)$, puisque $P(S, X) = P(S \mid X)P(X)$ et que, dans ce cadre, $P(X)$ est une constante.

Il n'est pas possible, en pratique, d'énumérer explicitement tous les chemins possibles pour choisir le plus probable : avec une séquence de longueur n et m états cachés, il y a m^n chemins possibles. L'algorithme de Viterbi permet de calculer récursivement le chemin s^* le plus probable, sans les énumérer tous.

Supposons que l'on connaisse, pour tous les états u, la probabilité du chemin le plus probable finissant dans l'état u pour l'observation partielle X_1^t

$$V_u(t) = max_{s_1^t} P(S_1^{t-1} = s_1^{t-1}, S_t = u, X_1^t = x_1^t).$$

Les termes correspondant pour la position $t+1$ peuvent alors se déduire des termes en t :

$$V_v(t+1) = b_v(x_{t+1}) \times max_u(V_u(t)a(u, v)).$$

En effet, pour calculer la probabilité du meilleur chemin finissant dans l'état v par la lettre x_{t+1}, il faut :

– choisir la meilleure façon d'arriver en $t+1$ (terme $max_u(V_u(x_t)a(u, v))$),

– puis émettre la lettre x_{t+1} conditionnellement à l'état v (terme $b_v(x_{t+1})$).

En gardant, à chaque position et pour chaque état caché, la mémoire du meilleur état précédent (le u sélectionné), la séquence des états cachés peut être reconstruite par cheminement inverse.

La formulation de l'algorithme est la suivante.

Initialisation : t=1,

$$V_u(1) = \pi(u) b_u(x_1)$$

Récurrence : t=2 à n,

$$V_v(t) = b_v(x_t) \times max_u(V_u(t-1)a(u,v))$$

$$ptr_t(v) = argmax_u(V_u(t-1)a(u,v))$$

Le pointeur $ptr_t(v)$ mémorise l'état précédent, u, qui permet d'obtenir la probabilité maximale.

Terminaison : l'état qui permet d'obtenir la meilleure probabilité au temps n est sélectionné,

$$s_n^* = argmax_u(V_u(n))$$

Cheminement inverse : pour t décroissant de $n-1$ à 1, on va rechercher en arrière le meilleur état précédent,

$$s_t^* = ptr_t(s_{t+1}^*)$$

Remarque : l'algorithme de Viterbi est un algorithme de programmation dynamique analogue à celui utilisé pour trouver l'alignement optimal entre deux séquences. Il s'agit ici d'aligner la séquence observée avec la séquence d'états cachés qui maximise la probabilité conjointe de la séquence d'états cachés et de la séquence observée.

Algorithme forward/backward

L'algorithme forward/backward permet de calculer, en chaque site de la séquence, la probabilité associée à chacun des états cachés u sachant la séquence observée :

$$P(S_t = u \mid X).$$

Ces probabilités sont appelées probabilités *a posteriori*. Elles sont calculées au moyen d'une récurrence avant/arrière dont l'algorithme tient son nom.

La **récurrence avant** consiste à calculer les termes $P(S_t = u \mid x_1^{t-1})$, probabilités d'être dans l'état caché u à la position t sachant la séquence observée jusqu'en $t-1$, et $P(S_t = u \mid x_1^t)$, probabilités d'être dans l'état caché u à la position t sachant la séquence observée jusqu'en t.

Initialisation : $t = 1$,

$$P(S_1 = u) = \pi(u)$$

Récurrence et terminaison : t=2 à n, équation *prédictive* :

$$P(S_t = v \mid x_1^{t-1}) = \sum_u a(u, v) P(S_{t-1} = u \mid x_1^{t-1})$$

Pour calculer la probabilité d'être dans l'état caché v à la position t sachant la séquence observée jusqu'en $t-1$, il faut tenir compte de toutes les façons possibles d'atteindre l'état v, d'où la somme sur les états précédents u. Cette équation est appelée *équation prédictive* car elle calcule la probabilité de l'état caché à la position t alors que la séquence observée n'est pas encore connue en t.

Les termes $P(S_{t-1} = u \mid x_1^{t-1})$ sont calculés par l'équation dite de *filtrage* :

$$P(S_t = u \mid x_1^t) = \frac{b_u(x_t) P(S_t = u \mid x_1^{t-1})}{\sum_v b_v(x_t) P(S_t = v \mid x_1^{t-1})}$$

La **récurrence arrière** consiste à calculer les termes $P(S_t = u, S_{t+1} = v \mid x_1^n)$, probabilités d'être dans l'état u à la position t et dans l'état v à la position $t+1$ sachant la séquence observée pour en déduire les termes $P(S_t = u \mid x_1^n)$, probabilités d'être dans l'état u sachant la séquence observée.

Initialisation : t=n,

$P(S_n = u \mid x_1^n)$ est donné par l'équation de filtrage au temps n.

Récurrence et terminaison : t=$n-1$ à 1, par l'équation de *lissage* ,

$$P(S_t = u, S_{t+1} = v \mid x_1^n) = \frac{P(S_t = u \mid x_1^t) a(u, v) P(S_{t+1} = v \mid x_1^n)}{P(S_{t+1} = v \mid x_1^t)}$$

ce qui permet de calculer

$$P(S_t = u \mid x_1^n) = \sum_v P(S_t = u, S_{t+1} = v \mid x_1^n)$$

Prédiction du chemin caché par l'algorithme forward/backward

Le chemin caché peut ainsi être reconstruit en choisissant, à chaque position dans la séquence, l'état qui a la plus forte probabilité *a posteriori*. Il est possible que cette reconstruction fournisse un chemin caché incohérent avec le modèle, si certaines transitions entre états cachés ne sont pas autorisées. Si nous reprenons l'exemple de la figure 3.1, la prédiction par forward/backward peut nous amener à prédire l'état rouge à un site et l'état jaune au site suivant, alors que cette transition n'est pas autorisée dans le graphe d'états. L'avantage, en revanche, est que l'on dispose des probabilités associées à chaque état caché. Il existe des palliatifs à ce problème [93].

Justifications des équations de l'algorithme forward/backward

Les justifications des équations prédictive, de filtrage et de lissage utilisent :

- Le théorème de Bayes qui permet d'écrire, pour deux évènements A et B,

$$P(A \mid B) = \frac{P(A, B)}{P(B)}$$

- Les propriétés d'indépendance conditionnelle des HMM (voir par exemple [134]). Notons simplement que :
  - l'observation x_1^{t-1} et l'état S_t sont indépendants conditionnellement à S_{t-1} :

$$P(S_t = v \mid S_{t-1} = u, x_1^{t-1}) = P(S_t = v \mid S_{t-1} = u)$$

  - de même, l'observation x_t^n et l'état S_{t-1} sont indépendants conditionnellement à S_t :

$$P(S_{t-1} = u \mid S_t = v, x_1^n) = P(S_{t-1} = u \mid S_t = v, x_1^{t-1})$$

  - les observations x_t et x_1^{t-1} sont indépendantes conditionnellement à S_t :

$$P(x_t \mid S_t = u, x_1^{t-1}) = P(x_t \mid S_t = u)$$

    En cas de mémoire d'ordre l, il faudrait bien sûr écrire $P(x_t \mid S_t = u, x_1^{t-1}) = P(x_t \mid S_t = u, x_{t-l}^{t-1})$.

Equation prédictive

$$\begin{aligned}
P(S_t = v \mid x_1^{t-1}) &= \sum_u P(S_{t-1} = u, S_t = v \mid x_1^{t-1}) \text{ on somme sur tous les états précédents} \\
&= \sum_u P(S_t = v \mid S_{t-1} = u, x_1^{t-1}) P(S_{t-1} = u \mid x_1^{t-1}) \text{ théorème de Bayes} \\
&= \sum_u P(S_t = v \mid S_{t-1} = u) P(S_{t-1} = u \mid x_1^{t-1}) \text{ indépendance conditionnelle} \\
&= \sum_u a(u,v) P(S_{t-1} = u \mid x_1^{t-1})
\end{aligned}$$

Equation de filtrage

$$\begin{aligned}
P(S_t = u \mid x_1^t) &= P(S_t = u \mid x_1^{t-1}, x_t) \text{ } x_1^t \text{ est décomposé en } x_1^{t-1} x_t \\
&= \frac{P(S_t = u, x_t \mid x_1^{t-1})}{P(x_t \mid x_1^{t-1})} \text{ théorème de Bayes} \\
&= \frac{P(S_t = u, x_t \mid x_1^{t-1})}{\sum_u P(S_t = u, x^t \mid x_1^{t-1})} \text{ somme sur les } u \text{ au dénominateur} \\
&= \frac{P(S_t = u \mid x_1^{t-1}) P(x_t \mid S_t = u, x_1^{t-1})}{\sum_u P(S_t = u \mid x_1^{t-1}) P(x_t \mid S_t = u, x_1^{t-1})} \\
&= \frac{P(S_t = u \mid x_1^{t-1}) P(x_t \mid S_t = u)}{\sum_u P(S_t = u \mid x_1^{t-1}) P(x^t \mid S_t = u)} \text{ indépendance conditionnelle} \\
&= \frac{b_u(x_t) P(S_t = u \mid x_1^{t-1})}{\sum_u b_u(x_t) P(S_t = u \mid x_1^{t-1})}
\end{aligned}$$

Equation de lissage

$$\begin{aligned}
P(S_t = u, S_{t+1} = v \mid x_1^n) &= P(S_t = u \mid S_{t+1} = v, x_1^n) P(S_{t+1} = v \mid x_1^n) \text{ théorème de Bayes} \\
&= P(S_t = u \mid S_{t+1} = v, x_1^t) P(S_{t+1} = v \mid x_1^n) \text{ indépendance conditionnelle} \\
&= \frac{P(S_t = u, S_{t+1} = v \mid x_1^t) P(S_{t+1} = v \mid x_1^n)}{P(S_{t+1} = v \mid x_1^t)} \text{ théorème de Bayes} \\
&= \frac{P(S_t = u \mid x_1^t) P(S_{t+1} = v \mid S_t = u, x_1^t) P(S_{t+1} = v \mid x_1^n)}{P(S_{t+1} = v \mid x_1^t)} \\
&= \frac{P(S_t = u \mid x_1^t) P(S_{t+1} = v \mid S_t = u) P(S_{t+1} = v \mid x_1^n)}{P(S_{t+1} = v \mid x_1^t)} \text{ ind. cond.} \\
&= \frac{P(S_t = u \mid x_1^t) a(u,v) P(S_{t+1} = v \mid x_1^n)}{P(S_{t+1} = v \mid x_1^t)}
\end{aligned}$$

3.1.3 Estimation des paramètres

Intéressons nous maintenant à la question de l'estimation des paramètres. L'estimation des paramètres d'un HMM dépend des informations disponibles sur la séquence à modéliser. Selon les cas, les données seront qualifiées de :

- annotées : les états cachés sont connus le long de la séquence utilisée pour estimer les paramètres,
- non annotées : les états cachés ne sont pas connus,
- étiquetées : les états cachés ne sont pas connus en tant que tels, mais une information partielle est disponible, sur le *type* d'état caché par exemple.

Estimation à partir de données annotées

Dans le cas de données annotées, le processus caché est connu pour les séquences d'apprentissage. Par exemple, si chaque type de structure secondaire est modélisé par un état caché, la suite des états cachés est disponible d'après les assignations de structures secondaires de l'ensemble d'apprentissage. Les paramètres a et b sont estimés par les formules :

$$\hat{a}(u,v) = \frac{N(u,v)}{\sum_{v\in\mathcal{S}} N(u,v)}$$
$$\hat{b}_u(x) = \frac{N(x \mid u)}{\sum_{y\in\mathcal{X}} N(y \mid u)}.$$

Si la loi initiale est prise identique à la loi stationnaire des états cachés, ces paramètres sont estimés par :

$$\hat{\pi}(u) = \frac{N(u)}{\sum_{v\in\mathcal{S}} N(v)}$$

$N(u)$ désigne le nombre d'occurrences de l'état caché u. $N(u,v)$ désigne le nombre d'occurrences de l'état u suivi de l'état v. $N(x \mid u)$ désigne le nombre d'occurrences de la lettre x dans l'état caché u. Ces estimateurs sont ceux qui maximisent la vraisemblance des données complètes $P(X,S)$. Ils aboutissent à un résultat tout à fait intuitif : les probabilités sont estimées par les fréquences observées. Cette procédure est appelée *estimation par comptage*.

Pour l'utilisation d'un ordre plus élevé sur le processus observé, il faut tenir compte du mot w précédent x. La formule d'estimation devient

$$\hat{b}_u(x;w) = \frac{N(wx \mid u)}{\sum_{y \in \mathcal{X}} N(wy \mid u)}$$

où $N(wx \mid u)$ désigne le comptage du mot wx finissant dans l'état caché u.

Estimation à partir de données non annotées : l'algorithme EM

Ce cas est le plus compliqué, mais il est aussi le plus intéressant, car il permet d'extraire des connaissances sur les séquences. En effet, dans le cas de données annotées, les plages à caractériser sont déjà connues. Dans le cas des données non annotées, elles vont être découvertes par le modèle.

L'estimation des paramètres se fait par maximisation de la vraisemblance, pour déterminer les paramètres $\hat{\theta}$ qui maximisent $P_\theta(X)$:

$$\hat{\theta} = argmax_\theta P_\theta(X).$$

L'algorithme EM (*Expectation Maximization*) est l'un des moyens d'approcher les estimateurs des paramètres a et b.

Si le chemin caché était connu, il serait possible d'appliquer les formules de l'estimation par comptage et de maximiser ainsi $P(X, S)$. Comme le chemin caché est inconnu, la procédure consiste à explorer (implicitement) tous les chemins possibles et à remplacer les comptages par leurs *espérances* sur tous les chemins cachés, c'est à dire les comptages attendus dans la séquence observée. Les espérances des comptages sont calculées grâce à l'algorithme forward/backward. Or ces calculs font intervenir les paramètres a et b, que l'on cherche justement à estimer.

Pour résoudre ce problème, la procédure employée est itérative : d'après un jeu initial de paramètres, les espérances des comptages sont calculées (étape E), puis elles sont réutilisées pour estimer un nouveau jeu de paramètres (étape M). Cet aspect itératif de l'algorithme est illustré dans la figure 3.4.

L'étape E consiste à calculer :

- $n^{(i)}(u, v) = E(N^{(i)}(u, v) \mid X)$, espérance conditionnelle du comptage $N(u, v)$ dans la séquence X d'après les paramètres de l'itération i,
- $n^{(i)}(x \mid u) = E(N^{(i)}(x \mid u) \mid X)$, espérance conditionnelle du comptage $N(x \mid u)$ dans la séquence X d'après les paramètres de l'itération i.

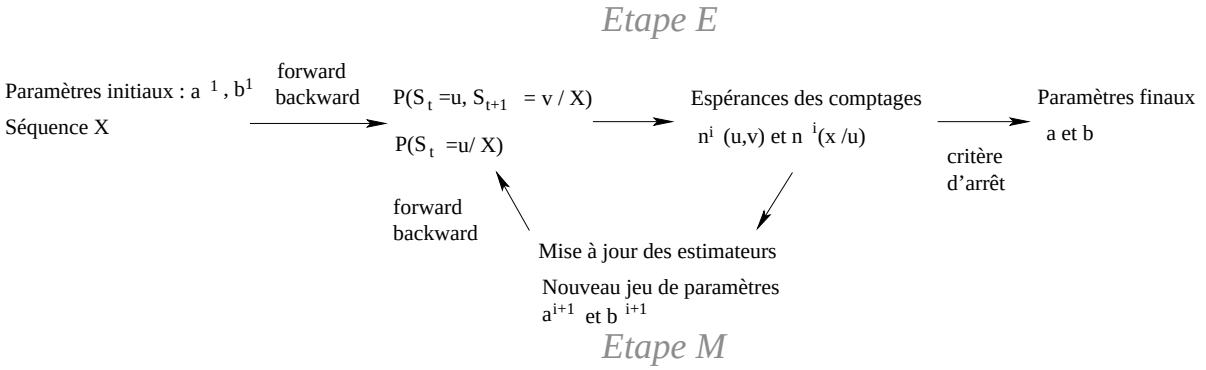

FIG. 3.4 – Illustration de l'algorithme EM pour l'estimation des paramètres d'un HMM à partir de données non annotées.

Dans le cadre des données annotées, les comptages peuvent s'écrire

$$N(u,v) = \sum_t I_{\{S_t=u, S_{t+1}=v\}},$$

$$N(x \mid u) = \sum_t I_{\{x_t=x\}} I_{\{S_t=u\}},$$

I étant une fonction indicatrice qui vaut 1 si la condition entre accolades est réalisée, et 0 sinon. En passant à l'espérance, ces indicatrices deviennent des probabilités :

$$n^{(i)}(u,v) = \sum_t P(S_t = u, S_{t+1} = v \mid x_1^n)$$

$$n^{(i)}(x \mid u) = \sum_t I_{\{x_t=x\}} P(S_t = u \mid x_1^n).$$

Les probabilités nécessaires à ces calculs sont fournies par l'algorithme forward/backward. L'exposant (i) indique que l'algorithme forward/backward utilise les valeurs courantes des estimateurs $\hat{a}$ et $\hat{b}$ à l'itération i. A nouveau, si l'ordre du processus observé est plus élevé, l'espérance des comptages devient :

$$n^{(i)}(wx \mid u) = \sum_t I_{\{x_{t-l}^t = wx\}} P(S_t = u \mid x_1^n).$$

L'étape M consiste à mettre à jour les paramètres qui seront utilisés à la prochaine itération $i+1$:

$$\hat{a}^{(i+1)}(u,v) = \frac{n^{(i)}(u,v)}{\sum_v n^{(i)}(u,v)}$$

$$\hat{b}_u^{(i+1)}(x) = \frac{n^{(i)}(x \mid u)}{\sum_{x'} n^{(i)}(x' \mid u)}$$

Il est démontré que cette procédure garantie la croissance de la vraisemblance de la séquence $P(X)$ [56, 132]. L'alternance des étapes E et M s'opère jusqu'à satisfaction d'un critère d'arrêt : soit un accroissement de la vraisemblance $P(X)$ jugé suffisamment faible, soit un nombre limité d'itérations.

L'algorithme EM est un algorithme de maximisation locale, il est donc soumis au problème d'optimum local : le maximum atteint dépend du point de départ (les valeurs initiales des paramètres). En pratique, ce problème peut être contourné en choisissant différents points de départ et en conservant le jeu de paramètres qui donne la meilleure vraisemblance.

Estimation à partir de données étiquetées CHMM

Les HMM pour données étiquetées ont été introduits par Krogh en 1994 [108] sous le nom de CHMM (class HMM). Ces modèles sont utilisés dans les cas où chaque plage -ou classe- est modélisée par un ensemble d'états cachés dont on souhaite estimer les transitions et lois d'émission des observations.

Prenons l'exemple d'un HMM pour les structures secondaires de protéines. Dans le cas le plus simple, chaque type de structure secondaire est modélisé par un état caché. Il est alors possible d'estimer les paramètres par comptage. Supposons maintenant que l'on souhaite modéliser chaque type de structure secondaire par plusieurs états cachés, dont on souhaite estimer les paramètres. Les données ne sont pas annotées en terme d'état caché, par contre elles sont annotées en terme de classe : le long de la séquence, la structure secondaire, autrement dit la classe d'appartenance des états cachés, est connue.

Un CHMM peut se formuler comme un HMM bi-dimensionnel dans lequel les états cachés émettent simultanément deux séquences : la séquence observée et la séquence des *étiquettes*, indiquant la classe modélisée. Les deux séquences sont émises indépendamment, conditionnellement au processus caché. La figure 3.5 présente le DAG d'un CHMM M1M0 adapté au cas des structures secondaires.

L'étiquetage des données ajoute au modèle les paramètres des lois d'émission relatives aux étiquettes : $b_u(e)$, paramètres des lois de probabilité associées à l'étiquette e pour l'état caché u. Dans le cas le plus simple, les étiquettes sont attribuées avec certitude : la loi de

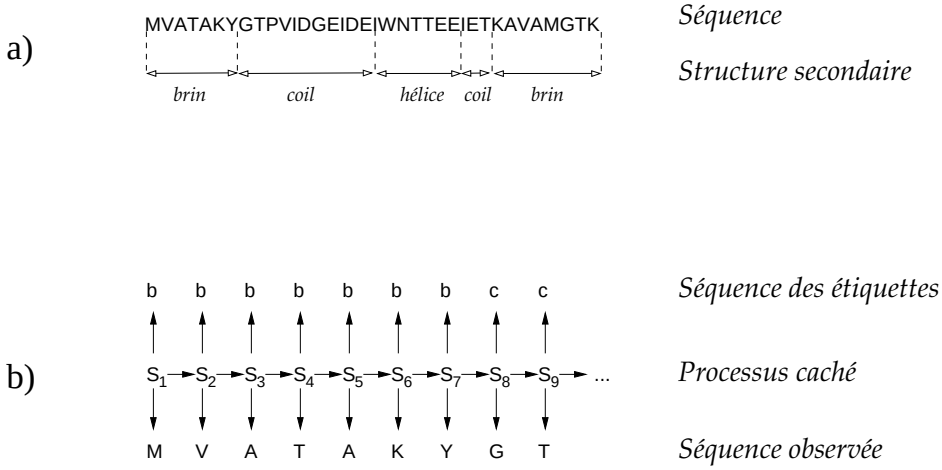

FIG. 3.5 – CHMM pour la structure locale des protéines. (a) Données étiquetées : la séquence de la protéine est annotée en terme de structure secondaire : hélices, brins, coil. (b) Modélisation CHMM : le processus caché émet simultanément la séquence de la protéine et la séquence des étiquettes de structures secondaires (h pour un résidu en conformation hélice, b pour brin et c pour coil).

probabilité est 1 pour une valeur de l'étiquette, et 0 pour toutes les autres. Cependant, il est possible de s'affranchir de cette certitude.

En pratique, l'estimation fait appel à l'algorithme EM. La séquence observée X et la séquence des étiquettes E étant émise indépendamment, conditionnellement au processus caché, il suffit de remplacer les termes $b_u(x)$ par le produit $b_u(x)b_u(e)$. Dans le cas où l'étiquetage est sans ambiguïté ($b_u(e) = 1$ pour une valeur de e et 0 pour toutes les autres), les $b_u(e)$ ne sont pas ré-estimés dans l'étape M.

3.1.4 Quelques généralisations des modèles de Markov

Le modèle de Markov suppose une dépendance à court terme, avec un ordre fixé. Rien n'empêche d'utiliser, dans un HMM, d'autres schémas de dépendance, pourvu qu'ils puissent se formuler sous forme d'une matrice de Markov classique.

Les modèles de mélanges de distribution de transition (MTD pour *Mixture Transition Distribution*) [19] sont des modèles dans lesquels les contributions des différents sites

dans le passé sont combinées additivement :

$$\begin{aligned} P(X_t = x_t \mid X_{t-l} = x_{t-l}, ..., X_{t-1} = x_{t-1}) &= \sum_{g=1}^{l} \lambda_g P(X_t = x_t \mid X_{t-g} = x_{t-g}) \\ &= \sum_{g=1}^{l} \lambda_g q_{x_{t-g} x_t} \end{aligned}$$

avec la contrainte $\sum_g \lambda_g = 1$. Les dépendances entre les X_t et les X_{t-g} sont exprimées sous forme d'une matrice markovienne Q, d'éléments $q_{x_{t-g} x_t}$.

Pour les séquences de protéines, un modèle MTD d'ordre l aura $19 \times 20 + l - 1$ paramètres (19×20 pour la matrice de Markov et $l - 1$ pour les λ_g). Ces modèles offrent une alternative aux modèles de Markov d'ordre élevé classiques, avec un nombre de paramètres moins importants. Les paramètres d'un modèle MTD sont estimés par maximisation numérique de la vraisemblance [18].

Il est possible, à partir des paramètres Q et λ d'un modèle MTD, de reconstituer la matrice markovienne d'ordre élevé correspondante. Ainsi, pour un modèle d'ordre 2 :

$$P(X_t = a \mid X_{t-2} = b, X_{t-1} = c) = \lambda_2 q_{ba} + \lambda_1 q_{ca}.$$

Une généralisation des modèles MTD consiste à utiliser différentes matrices Q pour chaque retard g.

Les chaînes de Markov à ordre variable [37] permettent d'adapter la taille de la mémoire au contexte de la séquence. La dépendance d'une lettre à son passé est exprimée grâce à un arbre de suffixe probabiliste, tel qu'illustré dans la figure 3.6.

Dans une chaîne de Markov classique, toutes les branches de l'arbre sont de même longueur. Dans le cas d'une chaîne de Markov à ordre variable, les branches ne sont pas nécessairement de la même longueur.

Les paramètres des lois d'émission d'une chaîne de Markov à ordre variable sont estimés par comptage, en partant de l'arbre maximal avec toutes les branches de même longueur, fixée au départ. L'arbre est ensuite *élagué* d'après un critère de comparaison des lois d'émission incluant ou non le noeud à élaguer. Ce critère fait aussi intervenir le nombre de comptages. Ceci à son importance pour nous car les comptages de mots dans les séquences de protéines peuvent être très faibles en raison de la taille élevée de l'alphabet.

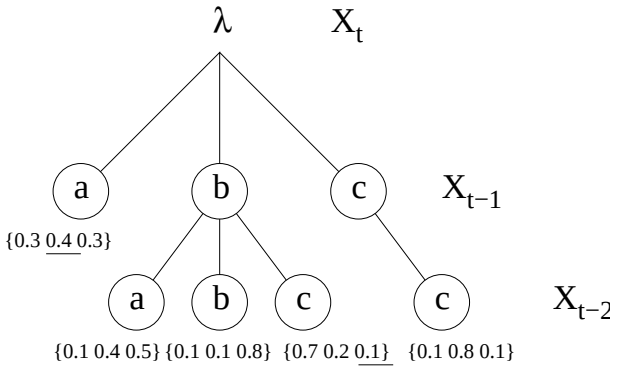

FIG. 3.6 – Arbre à suffixe probabiliste de profondeur 2 d'une séquence dans l'alphabet à 3 lettres abc. A chaque noeud de l'arbre est associée une loi d'émission pour la racine λ, conditionnellement au suffixe obtenu par le parcours dans l'arbre. Cette loi est représentée ici entre accolade, pour les 3 lettres a, b et c respectivement. Par exemple, pour cette séquence, $P(X_t = b \mid X_{t-1} = a) = 0.4$ et $P(X_t = c \mid X_{t-2} = c, X_{t-1} = b) = 0.1$.

Pour l'utilisation dans un HMM, la chaîne de Markov à ordre variable peut être représentée par une matrice de Markov maximale, dans laquelle plusieurs lignes sont égales, ici : $P(X_t \mid X_{t-2} = a, X_{t-1} = a) = P(X_t \mid X_{t-2} = b, X_{t-1} = a) = P(X_t \mid X_{t-2} = c, X_{t-1} = a) = P(X_t \mid X_{t-1} = a)$.

Les modèles de Markov *à trous* (*sparse Markov models*) sont des modèles d'arbres à suffixes dans lesquels des jokers (i.e., n'importe quelle lettre) sont autorisés [62]. Un tel arbre est représenté dans la figure 3.7.

Cette structure de dépendance peut être exprimée sous la forme d'une matrice de Markov maximale dans laquelle plusieurs éléments sont égaux.

Les modèles de Markov parcimonieux (PMM pour *Parcimonious Markov Model*) sont une généralisation des chaînes de Markov à ordre variable et des chaînes de Markov *à trous*, introduits par Bourguignon [25]. Des fusions de contextes sont autorisées, ce qui se traduit dans l'arbre à suffixe par des jokers partiels, comme illustré dans la figure 3.8.

L'estimation des paramètres d'un modèle PMM dépend de la taille de l'alphabet utilisé. Pour un alphabet de taille réduite, l'estimation part de l'arbre complet, contenant toutes les fusions possibles, et élague les branches par programmation dynamique [25]. Avec un alphabet plus grand, il est nécessaire de spécifier les fusions de contextes autorisées ou d'utiliser une approche d'échantillonnage.

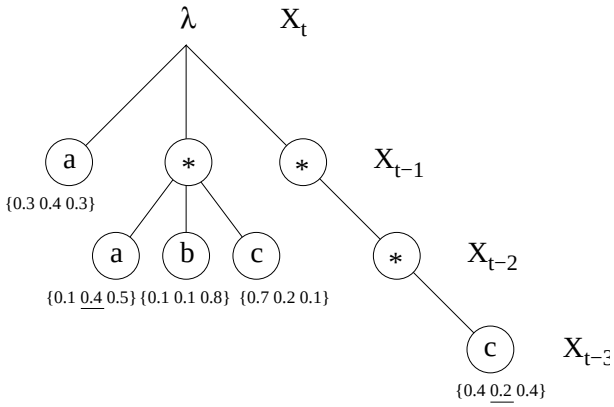

FIG. 3.7 – Arbre à suffixe probabiliste *à trous* de profondeur 3 d'une séquence dans l'alphabet à 3 lettres abc. * symbolise le joker : n'importe laquelle des 3 lettres. Ici $P(X_t = b \mid X_{t-2} = a, X_{t-1} = a) = P(X_t = b \mid X_{t-2} = a, X_{t-1} = b) = P(X_t = b \mid X_{t-2} = a, X_{t-1} = c) = P(X_t = b \mid X_{t-2} = a, X_{t-1} = *) = 0.4$. Plusieurs jokers successifs sont autorisés : $P(X_t = b \mid X_{t-3} = c, X_{t-2} = *, X_{t-1} = *) = 0.2$.

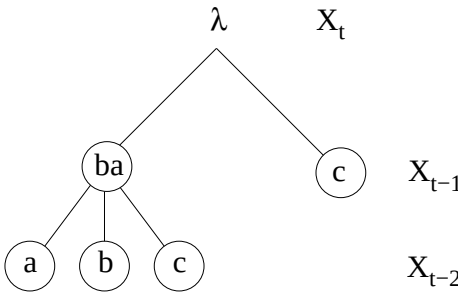

FIG. 3.8 – Arbre à suffixe probabiliste d'un PMM de profondeur 2 d'une séquence dans l'alphabet à 3 lettres abc. Un joker partiel est autorisé au niveau -1 : $P(X_t \mid X_{t-2} = a, X_{t-1} = a) = P(X_t \mid X_{t-2} = a, X_{t-1} = b)$.

Pour l'utilisation dans un HMM, les dépendances sont exprimées sous la forme d'une matrice de Markov maximale dont plusieurs lignes sont égales.

Les modèles de semi-Markov Les modèles de semi-Markov cachés (ou HSMM) permettent de modéliser explicitement le temps de séjour dans un état caché [149].

Dans un HMM, le temps de séjour dans un état caché suit une loi géométrique : si p est la probabilité de transition d'un état vers lui-même, la probabilité de rester n fois dans cet état est

$$P(\text{rester n fois dans l'état}) = p^{n-1}(1-p).$$

Cette distribution des temps de séjour peut ne pas être appropriée aux séquences modé-

lisées.

Dans un HSMM, la visite d'un état caché s'accompagne du tirage d'une durée de séjour d dans l'état, d'après une loi spécifique de l'état caché. d observations sont ensuite générées d'après la loi d'émission de l'état caché. L'état caché suivant est ensuite choisi d'après les probabilités de transitions entre états cachés. Les algorithmes de prédiction des HSMM ont une complexité de l'ordre du cube de la longueur de la séquence, et le nombre de paramètres à estimer est bien plus important que dans un HMM.

3.2 Applications de HMM à la prédiction de structure des protéines : état de l'art

La suite de ce chapitre présente les utilisations des HMM pour la prédiction de structure des protéines.

Les profils HMM permettent de fournir une prédiction de la structure globale dans le cas où une structure de protéine homologue est présente dans la PDB. Ils ne coïncident pas exactement avec le sujet de cette thèse, qui s'intéresse au problème de prédiction en l'absence de structure disponible. Néanmoins, leur utilisation très répandue justifie une brève présentation.

Les approches de prédiction de structure locale, notamment la structure secondaire, seront présentées plus en détail.

3.2.1 Prédiction de la structure globale des protéines par les profils HMM

Un profil HMM modélise un ensemble de séquences alignées. Un tel modèle est composé de trois types d'états :

1. des états *match* (M) modélisant les sites alignés d'un alignement multiple, avec des lois d'émission sur les acides aminés associées,
2. des états *insert* (I) modélisant les insertions, avec des lois d'émission sur les acides aminés associées,
3. des états *delete* (D) modélisant les délétions qui sont des états silencieux qui n'émettent pas de lettres.

La figure 3.9 représente un profil HMM correspondant à un alignement multiple de cinq séquences courtes dans l'alphabet acgt.

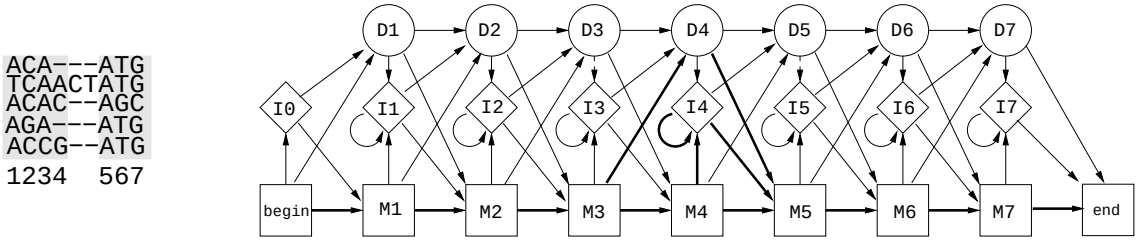

FIG. 3.9 – Exemple de profil HMM. A gauche, un alignement multiple de cinq séquences. Les parties grisées mettent en évidence les colonnes dans lesquelles plus de la moitié des séquences portent une lettre, ici choisies comme étant associées à des états *match*. A droite, le profil HMM correspondant. Il a une longueur de 7 sites. Les états *delete* silencieux sont symbolisés par des cercles, les états *match* par des carrés, et les états *insert* par des losanges. Ces derniers états sont les seuls à comporter des transitions sur eux-mêmes, pour modéliser des insertions de plusieurs résidus. Le début et la fin de l'alignement sont modélisés explicitement par les états silencieux *begin* et *end*. Dans ce cas très simple, certaines colonnes de l'alignement comportent uniquement des sites alignés. Certains états ne sont donc pas visités par l'alignement multiple dans le profil. Les transitions effectivement utilisées sont représentées par des flèches en gras.

Le paramétrage et l'utilisation de tels modèles sont décrits en détail dans [56]. A partir d'un alignement multiple, les paramètres d'un profil HMM peuvent être estimés par comptage, après avoir décidé de la taille du modèle. Celle-ci peut-être choisie égale au nombre de colonnes dans lesquelles plus de la moitié des séquences portent une lettre, comme dans l'exemple de la figure 3.9. L'apport de ces modèles est le cadre probabiliste introduit pour traiter les insertions/délétions.

Il est plus intéressant d'apprendre le modèle à partir de séquences non alignées, car cette procédure permet en plus de créer l'alignement. Krogh et al proposent ainsi une procédure d'apprentissage d'après des séquences non alignées, durant laquelle la taille du modèle est adaptée d'après les chemins empruntés dans le modèle [109]. Une fois les modèles construits, un score de vraisemblance est calculé entre la séquence à prédire et tous les profils, au moyen de l'algorithme forward/backward. Ceci nécessite le choix d'un modèle nul alternatif [16, 97]. L'alignement de la séquence avec le profil est obtenu par l'algorithme de Viterbi.

Les profils HMM permettent de détecter des homologues lointains, de manière plus efficace que les algorithmes d'alignement de séquences classiques comme PSI-BLAST [124]. Les logiciels SAM [98] et HMMER [57] sont dédiés à la construction et à l'utilisation des

profils HMM. La banque d'alignements de familles de protéines PFAM rassemble une collection de profils HMM construits d'après des alignements multiples [17].

Les profils HMM ont donné lieu à des développements méthodologiques particuliers pour l'estimation des paramètres [33, 96, 187, 2] et l'alignement de deux profils entre eux [58]. Un développement méthodologique récent, proposé par Karchin et al [95], consiste à former un profil HMM bidimensionnel. Les états *match* et *insert* d'un tel HMM émettent simultanément la séquence en acides aminés et la structure locale de la protéine. Ces deux séquences sont indépendantes, conditionnellement à la séquence cachée. La structure locale est décrite par la structure secondaire ou par un alphabet structural. Le HMM est entraîné sur les séquences de protéines et les probabilités associées aux structures locales prédites, d'après les séquences, par un réseau de neurones. Les mêmes informations sont fournies en entrée pour les séquences à prédire. L'information de structure locale ainsi utilisée permet d'améliorer la reconnaissance de repliement et l'alignement résultant.

3.2.2 Prédiction de la structure locale des protéines par les HMM

Différentes approches ont été proposées dans la littérature :

- une modélisation dite *automatique* par Asai et al en 1993 [6],
- une modélisation *experte* des protéines, spécialisée pour certaines classes par Stultz et White en 1993[186, 200],
- plus récemment, une approche basée sur une collection de fragments : HMMSTR par Bystroff et al en 2000 [36],
- la prise en compte d'une fenêtre glissante de structure secondaire par Crooks et Brenner [46], ainsi que Zheng [209], en 2004.

D'autres travaux n'utilisent pas de modélisation particulièrement élaborée [59]. Les modèles de semi-Markov cachés ont également été appliqués en prédiction de structure des protéines [171, 42, 10].

Modélisation *automatique* des structures secondaires par Asai et al, 1993 [6]

L'une des premières utilisations des HMM pour prédire les structures secondaires de protéines est due à Asai et al, en 1993. Dans cette étude, les auteurs entraînent séparément

4 sous-modèles HMM sur les séquences en hélices, brins, coudes et autres, au moyen de l'algorithme EM. Ce sont des modèles M1M1 : les lois d'émission des acides aminés tiennent compte des paires d'acides aminés. Plusieurs topologies sont testées pour chaque sous-modèles, celles qui discriminent le mieux chaque classe sont sélectionnées. Les sous-modèles sont ensuite réunis pour former le HMM complet, montré dans la figure 3.10.

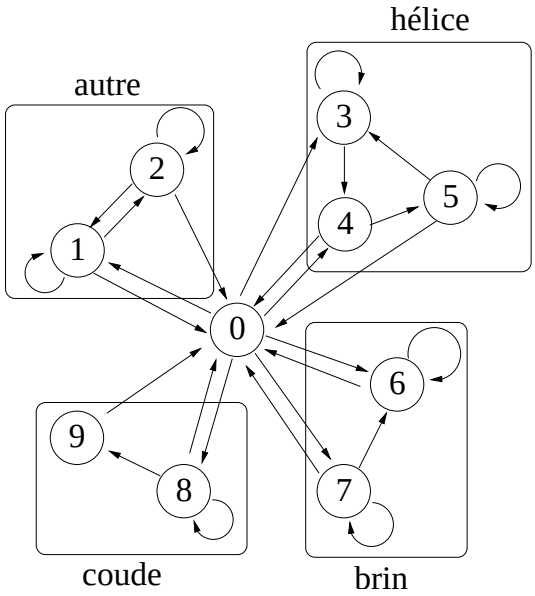

FIG. 3.10 – HMM proposé par Asai et al pour la prédiction de structure secondaire [6]. Dans ce modèle, les émissions d'acides aminés sont associées aux **transitions** entre états.

Le score Q_3 pour la prédiction en 3 classes hélice/brin/autre+turn (pourcentage de résidus correctement prédits) est de 54.7 %. Les différents sous-modèles testés (nombre d'états cachés, transitions entre états) ne sont pas précisés.

Modélisation experte de classes de protéines, par White et Stultz, 1993 [186, 200].

Cette approche s'apparente à l'utilisation des profils HMM. D'après des alignements de séquences de protéines, dont l'une de structure connue expérimentalement, Stultz et al proposent une collection de HMM représentant l'architecture interne de différentes classes de structures 3D.

Cette collection de HMM est utilisée pour prédire la classe structurale d'une séquence, puis sa structure secondaire.

Modèles proposés La publication initiale mentionne une collection de modèles pour 15 classes structurales.

Les sites des protéines sont décrits par 13 états structuraux : résidus terminaux en N-ter et C-ter et coeur d'hélice α, hélice α exposée et enfouie, 4 états pour les coudes β, brin β enfoui, exposé et intermédiaire et coil. Les modèles sont construits en utilisant ces états structuraux, d'après une structure extraite de la PDB représentative pour chaque classe et des séquences alignées. La figure 3.11 représente un modèle HMM de la classe structurale α/β contenant un feuillet β constitué de 5 à 7 brins, d'après la publication de 1993 [186].

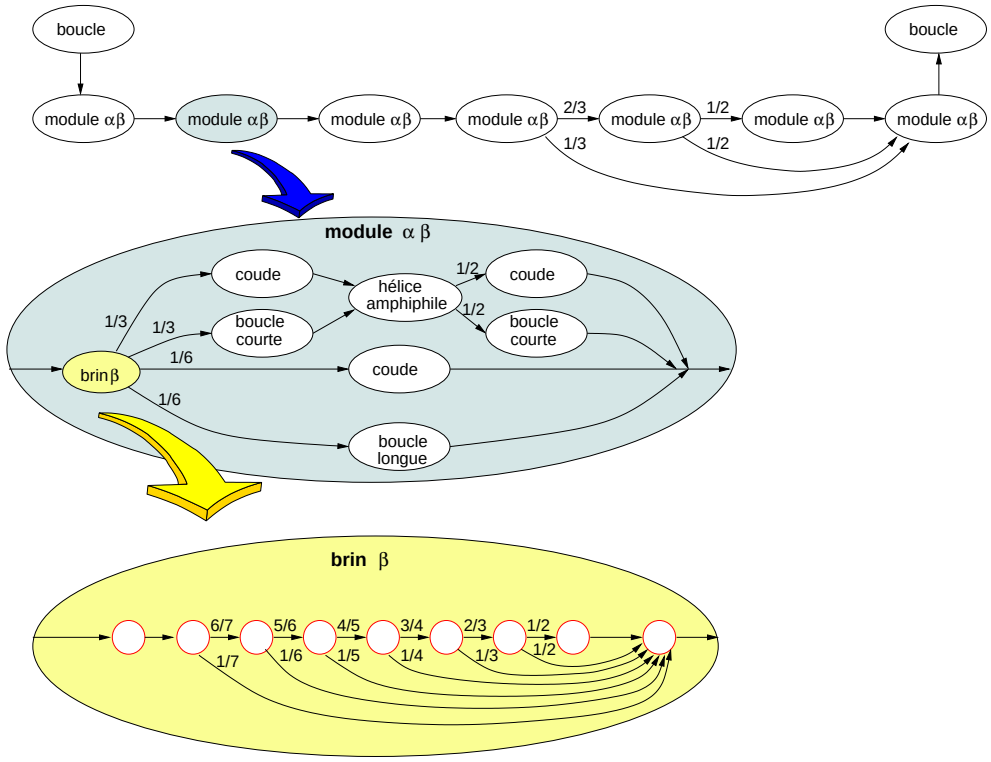

FIG. 3.11 – HMM d'une protéine de la classe α/β contenant un feuillet de 5 à 7 brins, d'après [186]. La partie supérieure de la figure illustre l'organisation générale de la protéine en modules structuraux. La partie médiane détaille la composition d'un module structural en segments de structures secondaires. La partie inférieure explicite le modèle HMM d'un brin β long de 3 à 9 résidus. Les cercles rouges représentent les états cachés.

De 4 à 25 HMM sont proposés pour chaque classe, pour modéliser différentes gammes de longueurs de protéines.

Estimation des paramètres Les paramètres sont estimés par comptage, ce qui suppose que les séquences soient annotées en terme d'états structuraux.

Pour chaque modèle, la séquence de la protéine à modéliser -de structure connue- est alignée avec des séquences homologues. Le nombre de séquences alignées ne dépasse pas 20 dans la publication de 1993 [186]. Seuls les acides aminés distincts sont pris en compte pour le calcul des fréquences (une lettre qui apparaît plusieurs fois n'est comptée qu'une fois) afin de prévenir le biais introduit par des homologues très proches. En effet, plusieurs séquences très proches, conduiraient à des fréquences biaisées [2]. Il semble que ces fréquences soient utilisées pour calculer les lois d'émission associées aux 13 états structuraux. En d'autres termes, tous les états cachés de même type structural seraient caractérisés par les mêmes lois d'émission. Ce point n'est pas clairement explicité.

Remarque : Si tel est le cas, cette contrainte d'égalité entre lois d'émission est une contrainte utile pour estimer les paramètres, mais sans doute inappropriée, en pratique, dans certains cas. En effet, cette procédure *moyenne* plusieurs lois différentes qui peuvent être assez informatives. Par exemple, les lois d'émission sont les mêmes en début et en fin d'hélice.

Il semble que certaines probabilités de transitions sont fixées arbitrairement (par exemple les probabilités associées aux *sauts* de modules α/β sur la figure 3.11), sans être ré-estimées. Une estimation des paramètres à partir de ces valeurs initiales nous semblerait plus raisonnable.

Prédiction Dans un premier temps, la séquence à analyser est assignée à une classe d'après la vraisemblance de la séquence sous chaque modèle. Le modèle le plus vraisemblable est sélectionné. Les structures secondaires sont prédites grâce à ce HMM, par les probabilités de l'algorithme forward/backward, en sommant les probabilités associées aux états cachés représentant la même structure secondaire. La première étape de classification est analogue à la reconnaissance de repliement par les profils HMM. Pour cette raison, cette méthode ne peut pas fournir de prédiction correcte sur des séquences qui ne sont pas représentées dans la collection de HMM. En ce qui concerne l'étape de prédiction des structures, dans la publication de 1993, l'évaluation n'est faite que sur deux protéines appartenant à la même classe structurale, dont l'une fait partie de l'ensemble d'apprentissage. Les scores Q_3 sont de 66 et 77%.

[2] des méthodes d'estimation ont été proposées dans le cas des profils HMM, pour prévenir les biais dus à la représentativité des séquences dans une famille et à la faible quantité de données, voire par exemple [33, 96, 56, 2]

Evolution de la méthode Un serveur web est aujourd'hui disponible à l'url `http://bmerc-www.bu.edu/psa/` : PSA (Protein Structure Analysis). Le nombre de classes structurales modélisées est étendu à 24, avec 1 à 54 modèles par classes. D'autre part, des modèles dits *génériques* de protéines globulaires et de protéines membranaires sont proposés pour fournir une prédiction sur les séquences non prises en compte par les autres modèles. Le modèle générique des protéines globulaires n'est pas détaillé. Les hypothèses de modélisation affichées sont minimales (`http://bmerc-www.bu.edu/psa/dsm-type-2.htm#dsm_type2`) :

- les hélices et les brins sont définis comme étant équi-probablement enfouis ou exposés,
- les coudes sont définis comme équi-probablement longs de 2 ou 4 résidus,
- les longueurs moyennes de segments de structures secondaires sont en accord avec les distributions de longueur observée.

Ces hypothèses nous semblent relativement grossières. Les modèles génériques n'ont pas été publiés, et le site ne donne pas plus de détail sur leurs structures. Nous n'avons pas connaissance d'une évaluation récente de ce serveur de prédiction sur des jeux de données de taille raisonnable. Nous ne pouvons donc pas juger de la qualité de ces modèles.

Méthode PASSML, Goldman et al, 1996 [192, 77, 78, 121]

Goldman et al proposent une formulation HMM des structures secondaires, couplée à un modèle d'évolution, pour estimer la phylogénie et prédire les structures secondaires des séquences protéiques.

Modèle proposé L'architecture interne des protéines en termes de structures secondaires est modélisée par un HMM de type M1M0 à 38 états cachés [121]. 8 types d'états cachés sont distingués, selon la structure secondaire et l'accessibilité à l'eau : hélice enfouie, hélice exposée, brin enfoui, brin exposé, coude enfoui, coude exposé, coil enfoui et coil exposé. La modélisation à 38 états cachés permet de tenir compte des distributions de longueurs des éléments de structures secondaires. Les 38 états cachés sont répartis en 20 états pour les hélices, 12 états pour les brins, 4 pour les coudes et 2 pour les coils, également répartis dans chaque classe entre états exposés et états enfouis. Le nombre d'états cachés nécessaire pour modéliser chaque classe est choisi par un critère de vraisemblance

pénalisée portant sur l'accord entre les distributions de longueur observées et produites par le modèle.

La figure 3.12 présente le HMM des brins β.

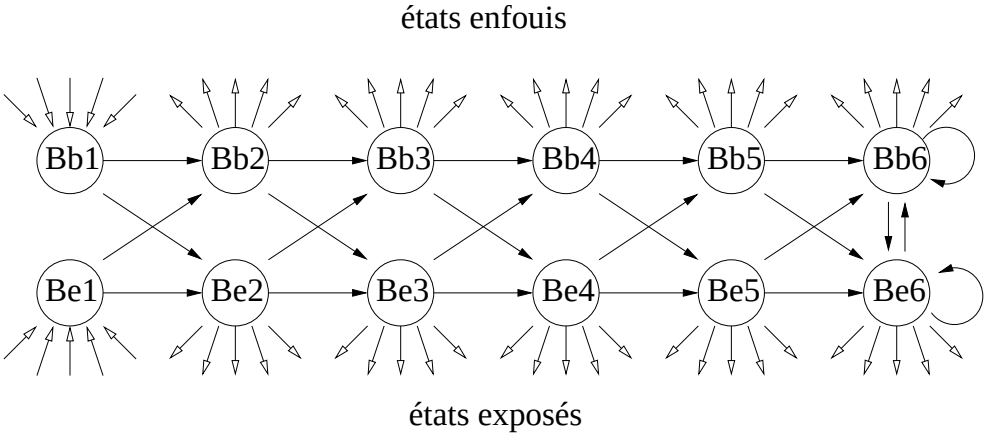

FIG. 3.12 – HMM d'un feuillet β d'après [121]. Les états enfouis (Bb) alternent avec les états exposés (Be). Les flèches creuses représentent les transitions vers les autres états du HMM. La contrainte de longueur minimale de 2 résidus est réalisée par l'absence de transitions des états Be1 et Bb1 vers l'extérieur du brin. Les états Bb6 et Be6 peuvent être visités plusieurs fois de suite.

Les hypothèses de modélisation sont les suivantes [121, 78] :

- Le caractère amphiphile des hélices et des brins (hélices et brins situés à la surface de la protéine, ayant une face hydrophobe et une face hydrophile) est pris en compte par l'alternance entre états cachés hydrophobes et hydrophiles.
- La progression dans le HMM est unidirectionnelle : l'entrée dans un brin ne peut se faire que dans les états Be1 ou Bb1, puis Be2 ou Bb2 ou vers l'extérieur.
- Seuls les derniers états cachés de chaque classe de structures secondaires peuvent être visités plusieurs fois (transitions sur eux-mêmes).
- Des longueurs minimales sont fixées à 4 résidus pour les hélices et 2 résidus pour les brins.

Estimation des paramètres Les probabilités de transition entre états cachés ne sont pas estimées directement par comptage, en raison de la faible quantité de données utilisée (207 protéines). Pour réduire le nombre de paramètres associés aux transitions, des hypothèses simplificatrices sont posées, permettant d'introduire des contraintes sur les paramètres :

– La probabilité de transition d'un état caché vers une autre classe structurale ne dépend pas de l'accessibilité de l'état sortant : $P(S_t = Bb2, S_{t+1} = \bar{B}) = P(S_t = Be2, S_{t+1} = \bar{B})$, $\bar{B}$ représentant l'extérieur du brin. Ces probabilités de sortie sont fixées de façon à respecter les distributions de longueurs observées.
– Les probabilités de rester dans les états terminaux (Bb6 et Be6) sont fixées de manière à reproduire les longueurs moyennes de segments de structures secondaires.
– La probabilité de transition entre deux états cachés de classes structurales différentes ne dépend pas de la position : $P(S_t = Bb2, S_{t+1} = Hb1) = P(S_t = Bb3, S_{t+1} = Hb1)$, $Hb1$ représentant le premier état enfoui d'une hélice. Ces probabilités sont estimées par comptage.
– La probabilités de transition entre états cachés de même classe structurale ne dépend pas de la position : $P(S_t = Bb2, S_{t+1} = Bb3) = P(S_t = Bb3, S_{t+1} = Bb4)$. Ces probabilités sont estimées par comptage.

Tous les états cachés de même type (structure secondaire et accessibilité) possèdent les mêmes lois d'émission d'acides aminés. Les lois d'émission des 8 types d'états cachés sont estimées par comptage. La remarque faite à propos de l'estimation des paramètres des modèles proposés par White et Stultz s'applique aussi à ces modèles.

Prédiction Le HMM est utilisé en couplage avec un modèle d'évolution. Le modèle d'évolution est un processus de saut markovien à sites indépendants, spécifique pour chaque type d'état caché. La probabilité pour un état caché de type k, de passer de l'acide aminé i à l'acide aminé j après un t est donné par :

$$\begin{cases} p_{ij}^k = \alpha_{ij}^k t & \text{si } i \neq j \\ p_{ii}^k = 1 - \sum_{j \neq i} \alpha_{ij}^k t & \text{sinon.} \end{cases}$$

Ceci est valable si la quantité d'évolution t séparant les deux séquences est suffisamment petite pour que la possibilité de substitution multiple puisse être négligée.

α_{ij}^k est le taux de remplacement instantané de i par j pour la catégorie k. Par définition $\alpha_{ij}^k = -\sum_{j \neq i} \alpha_{ij}$. Le processus de remplacement est réversible :

$$\pi_i^k \alpha_{ij}^k = \pi_j^k \alpha_{ji}^k,$$

où π_i^k est la probabilité stationnaire de l'acide aminé i dans l'état k. Les α sont estimés par

comptage pour chaque catégories d'états cachés d'après des paires de séquences proches alignées.

Ce système est utilisé pour estimer la topologie de l'arbre phylogénétique reliant des séquences alignées et leur structure secondaire commune. La topologie de l'arbre et les longueurs de branches sont estimées par maximisation numérique de la vraisemblance. La phylogénie peut être une donnée du problème, dans ce cas elle n'est pas estimée. La structure secondaire est estimée par les probabilités *a posteriori* des états cachés calculées par l'algorithme forward/backward. Cette étape nécessite le remplacement, dans les équations du forward/backward, des $b_u(x_t)$ -probabilités d'émission de la lettre x_t dans l'état caché u- par $b_u(\phi_t)$ -probabilités d'émettre l'ensemble des séquences alignées reliées par l'arbre phylogénétique donné, dans l'état caché u. Ceci est réalisé par la procédure décrite par Felsenstein [64].

Cette méthode permet donc d'estimer la phylogénie et de prédire la structure secondaire à partir d'un alignement de séquences. Les performances du modèle à 38 états en prédiction de structure n'ont pas été évaluées, à notre connaissance, sur des jeux de données de taille raisonnable. La publication de 1998 [121] comporte un graphique illustrant la prédiction sur une seule séquence, mais le score Q_3 n'est pas mentionné. Les auteurs rapportent seulement que « le programme prédit correctement une grande fraction des structures secondaires », et que les feuillets β sont mal prédits.

Dans les premier temps, la méthode de prédiction utilisait des HMM plus simples. Avec un HMM à 3 états cachés, l'évaluation de la prédiction est présentée, dans la publication de 1996 [77], sur une famille de 7 séquences. Sans tenir compte de la phylogénie, le Q_3 est de 65.7% sur la séquence seule, et la prédiction des brins β est déficiente. La prise en compte de la phylogénie (ici, estimée par la méthode) avec les 3 séquences les plus proches permet d'obtenir un Q_3 de 74.4% et améliore sensiblement la prédiction des brins. L'inclusion des 7 séquences fait baisser ce score : $Q_3 = 69.6\%$. Avec les mêmes informations, PHD fournit un Q_3 de 79.6%.

HMMSTR, Bystroff et al, 2000 [36]

La méthode HMMSTR propose une modélisation HMM des protéines basée sur la librairie de fragments I-sites. La collection de fragments I-sites a été obtenue par classification de fragments similaires en séquence, puis raffinement des classes sur des bases

structurales [34]. Initialement composée de 82 prototypes des 3 à 15 résidus, la librairie I-site a été enrichie en *masquant* les fragments correspondant à des I-sites, pour identifier des corrélations séquence/structure plus faibles [36], donnant lieu à une collection de 262 motifs. La figure 3.13 présente le I-site correspondant à une hélice amphiphile.

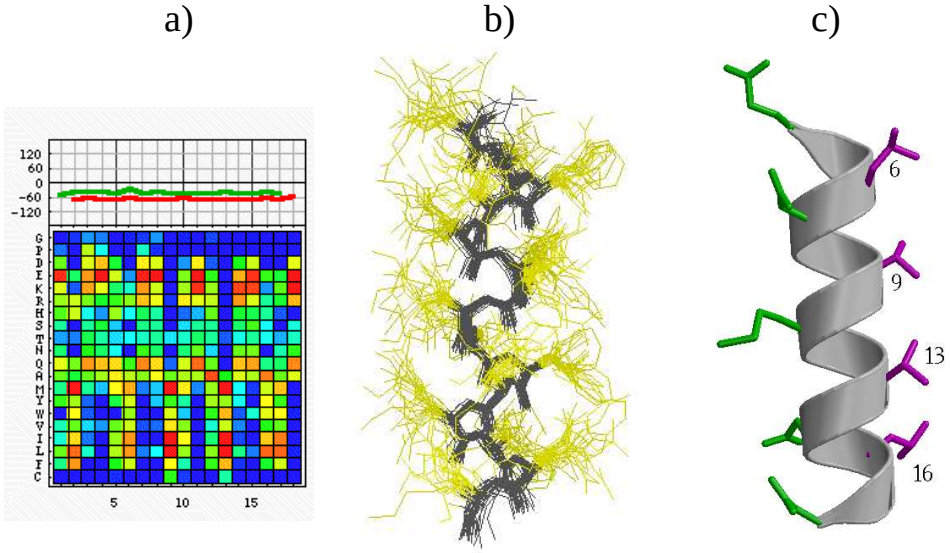

FIG. 3.13 – Motif I-site d'hélice amphiphile. a) en haut, graphique des angles Φ (rouge) et Ψ (vert) du prototype structural du I-site, en fonction de l'indice dans la séquence. En bas, fréquences normalisées des acides aminés à chaque position du motif : en rouge, la fréquence est trois fois plus élevée que la fréquence de base, en vert, les fréquences sont similaires et en bleu foncé la fréquence est trois fois moins élevée que la fréquence de base. b) superposition des 30 fragments les plus similaires. c) les chaînes latérales conservées sont mises en évidence, en vert pour les résidus polaires et en violet pour les résidus apolaires. Cette image provient du site web des I-sites http ://www.bioinfo.rpi.edu/applications/i-sites/Isites/.

Traduction des I-sites en HMM Les I-sites sont utilisés pour construire des HMM représentant l'architecture interne des protéines. Chaque I-site est *traduit* en petit HMM linéaire. Le nombre d'état cachés correspond à la longueur du motif. Ce HMM est de type M1M0 et bidimensionnel : il émet la séquence en acides aminés et une séquence de descripteurs structuraux. L'information de séquence est intégrée en terme de profils grâce à une distribution multinomiale. 3 types de descripteurs sont envisagés : les structures secondaires dans un alphabet à 3 lettres (hélice/brin/coil), les zones d'angles du diagramme de Ramachandran, dans un alphabet à 11 lettres (voir chapitre 2) et une description de contexte structural correspondant à une description en terme de structures secondaires enrichie, dans un alphabet à 9 lettres. Ce contexte structural tient compte des éléments

de structures secondaires encadrant les coudes et de l'appariement des brins β.

Initialisation du HMM complet La topologie du HMM est obtenue en opérant des fusion d'états cachés à partir de la collection de petits HMM. 2 critères sont proposés :

1. L'un est basé sur la co-occurrence des motifs I-sites au sein des protéines. Les I-sites sont prédits sur un jeu d'apprentissage. Les états cachés sont fusionnés sur la base de la fréquence de leur co-prédiction sur un même site de protéines. Un HMM de 208 états cachés est ainsi initialisé à partir des 2169 états cachés initiaux (correspondant à toutes les positions des 262 I-sites). Les probabilités de transitions sont estimées par comptage d'après un encodage des structures par les I-sites prédits avec le meilleur score de prédiction.
2. L'autre critère est basé sur l'alignement des profils des I-sites par programmation dynamique (la correspondance en terme de structure locale est contrôlée). Les paires d'états alignés sont fusionnés. Le HMM ainsi formé comporte 281 états cachés. Les probabilités de transitions initiales sont uniformes.

Dans les deux cas, un état silencieux est ajouté pour permettre aux parties disjointes du graphe de communiquer.

Estimation des paramètres Les paramètres sont estimés par l'algorithme EM, à partir d'un ensemble de protéines annotées en terme de descripteurs structuraux et de leurs profils PSI-BLAST.

Modification de la topologie du HMM Six heuristiques sont utilisées pour modifier la topologie du HMM durant l'estimation des paramètres :

- suppression de probabilités de transition trop faibles,
- suppression des états peu visités,
- fusion d'états,
- *re-distribution* des probabilités de transitions vers des transitions nulles,
- division d'un état en deux états presque identiques,
- ajout de transition entre deux états cachés.

Les trois premières fonctions permettent de simplifier le modèle, les trois dernières de la complexifier. Ces modifications sont suivies de ré-estimation des paramètres par l'EM.

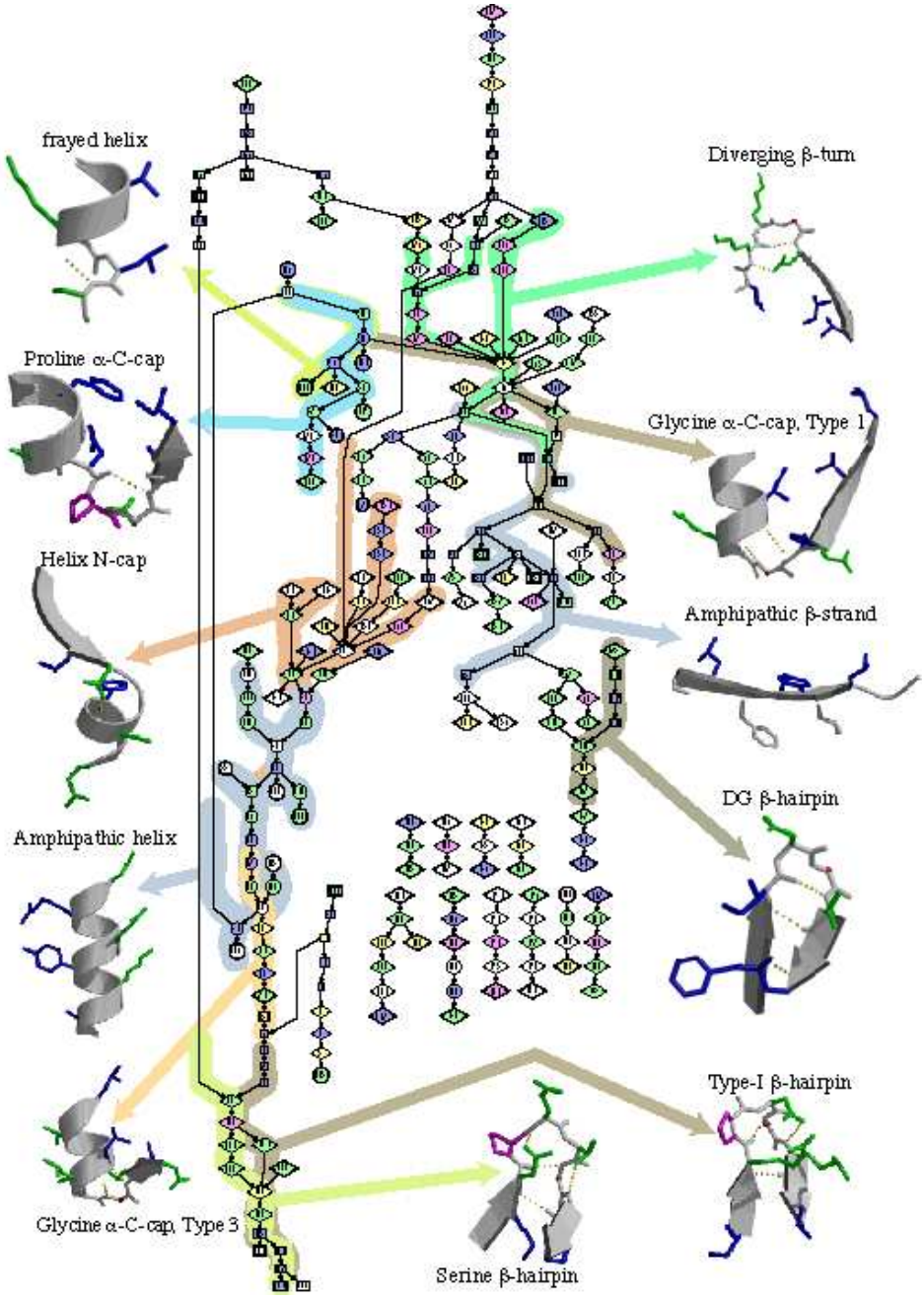

FIG. 3.14 – Modèle HMM pour la prédiction des zones d'angles Φ/Ψ. Ce modèle a 247 états. Les parties disjointes du graphe (au milieu) sont reliées au reste par un état silencieux, non montré ici. Certains chemins correspondant à des I-sites particuliers sont mis en évidence sur fond coloré. Cette image provient du site web des I-site http ://www.bioinfo.rpi.edu/applications/i-sites/Isites/.

La figure 3.14 présente le HMM obtenu par entraînement sur les zones d'angles Φ/Ψ et initialisation basée sur l'alignement des profils des I-sites. Ce modèle a l'avantage de permettre une interprétation intuitive : il illustre l'architecture interne des protéines en terme

de motifs structuraux dérivés des I-sites. En revanche, il est peu économe en paramètres.

Prédiction Ces modèles HMM sont utilisés pour prédire la structure locale des protéines grâce à l'algorithme forward-backward. La probabilité associée à une classe de structure secondaire est obtenue en sommant les probabilités d'émission de cette classe par les N états cachés, pondérées par les probabilités *a posteriori* des états cachés :

$$P(S_t = struct) = \sum_{n=u}^{N} P(S_t = u \mid profil) P(struct \mid u).$$

Les $P(S_t = u \mid profil)$ sont calculés par l'algorithme forward-backward et $P(struct \mid u)$ est la probabilité d'émission de la structure *struct* dans l'état caché u (i.e., un paramètre du modèle) ; la somme porte sur les N états cachés.

La score Q_3 obtenu est de 74.3%. Les résultats rapportés, très détaillés, montrent que prédiction est moins bonne pour les brins β, et que le modèle sous-prédit légèrement les brins et les hélices au profit du coil. Ces scores de prédiction sont satisfaisants, mais la taille du modèle est conséquente (nombre de paramètres non détaillé). La prédiction des zones d'angles est réalisée par une procédure de vote : une des trois grandes zones est choisie en sommant les probabilités en trois groupes, puis la zone de plus forte probabilité est prédite. Les performances de cette prédiction sont rapportées dans le chapitre 2.

Utilisation d'une fenêtre glissante dans la structure secondaire : les travaux de Crooks et Brenner, 2004 [46] et Zheng, 2004 [209]

Crooks et Brenner proposent une modélisation des protéines par un modèle HMM M1M0, dans lequel un état caché correspond à une fenêtre glissante dans la structure secondaire et l'observation correspondante est le résidu central de la fenêtre, comme illustré dans la figure 3.15.

Remarque Le modèle proposé par Crooks et Brenner modélise la structure secondaire par une chaîne de Markov d'ordre 1 sur des n-uplets de structures secondaires. Or un modèle de Markov d'ordre 1 sur des séquences recodées en n-uplets est un modèle de Markov d'ordre n sur les séquences non recodées. Cependant, il ne s'agit pas strictement d'un modèle MnM0 car l'utilisation des fenêtres de structure secondaire introduit une dépendance

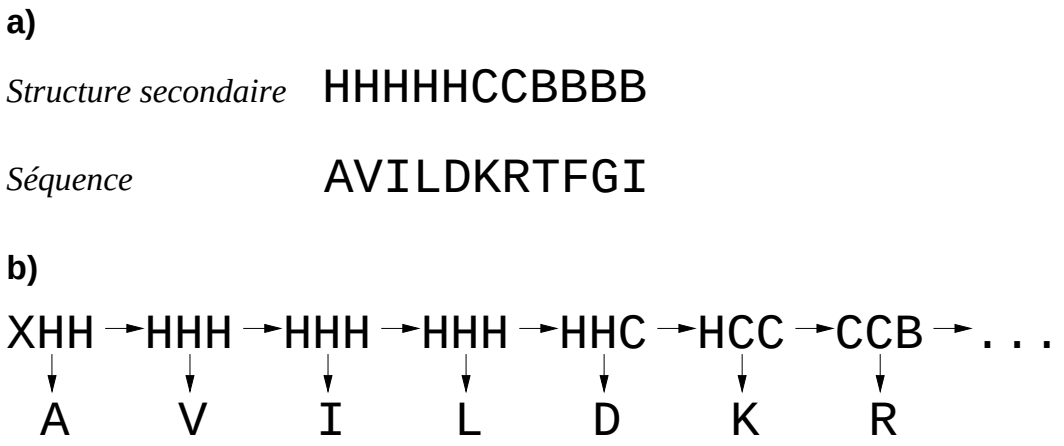

FIG. 3.15 – Modélisation HMM des protéines à l'aide d'une fenêtre glissante dans la structure secondaire. a) Données : structure secondaire (H pour hélice, B pour brin et C pour coil) et séquence de protéine). b) DAG de la modélisation HMM avec une fenêtre de taille 3 dans la structure secondaire. Les structures secondaires aux extrémités sont traitées comme indéterminées (X).

entre un n-uplet de structures secondaires et le résidu central de la fenêtre. Dans un modèle MnM0, les résidus sont indépendants, conditionnellement au processus caché d'ordre n : un acide aminé ne dépend que de la structure du résidu central, conditionnellement à la chaîne cachée.

L'information de séquence est utilisée sous forme de profils PSI-BLAST, avec un modèle de grandes déviations.

Avec une fenêtre de taille 7, le score Q_3 obtenu par ce modèle est de 66.4% avec les séquences seules et de 72.2% avec l'utilisation des profils. Il est difficile de donner une interprétation biologique à ce modèle.

La même approche avait été mise en œuvre par Delcher et al, en 1993 [50]. Les auteurs avaient introduit ce modèle sous le nom de *réseau causal*. En utilisant des doublets, le score Q_3 était de 62.2%.

Zheng utilise une approche du même type, en 2004 [209]. Les structures secondaires sont recodées en n-uplets. Les séquences sont également recodées en n-uplets. Le formalisme n'est pas totalement clair, le modèle semble être de type M1M0, un état caché étant un n-uplet de structures secondaires, qui émet un n-uplet d'acides aminés.

La taille de l'alphabet des protéines (20) ne permet pas d'estimer les probabilités de ces

n-uplets. Les acides aminés sont donc groupés, en utilisant un critère basé sur l'information mutuelle pour choisir les groupes.

Les meilleurs résultats sont obtenus avec un modèle sur les quintuplets de structures et les quintuplets d'acides aminés : score Q_3 de 67.9% avec l'algorithme forward/backward.

Thompson et Goldstein, 1997 [191] semblent, sans que cela soit explicitement mentionné, utiliser un modèle HMM M1M0 dans lequel un état caché est un n-uplet de descripteurs de la structure secondaire et de l'accessibilité au solvant. Les émissions des états cachés du modèles sont les colonnes des alignements multiples de la banque HSSP, modélisées par une loi multinomiale. Le profil émis ne correspond pas nécessairement au résidu central du n-uplet de descripteurs structuraux. Le score Q_3 atteint est de 71.6%, en utilisant les alignements de HSSP.

Autres travaux

Pour être exhaustif, citons aussi le travail de Edgoose et al [59], qui utilisent un modèle de type CHMM M1M0. Le choix du nombre d'états cachés du modèle est réalisé par un critère utilisant la théorie de l'information. Le modèle final a 6 états cachés, qui émettent indépendamment la séquence en acides aminés et la structure secondaire, conditionnellement au processus caché. Les lois d'émission des étiquettes de structure secondaire ne sont pas fixées. Ce modèle, testé sur 15 protéines seulement, donne un Q_3 de 62.2%.

Schmidler et al ont utilisé une modélisation des structures secondaires en terme de segments [171]. Contrairement à ce qui est mentionné dans l'article, le modèle utilisé est un HSMM. Il suppose l'indépendance entre les sites de segments différents, mais tient compte des corrélations au sein d'un même segment. Le score Q_3 atteint est de 68.8%, en utilisant les séquences seules.

Aydin et al [10], en reprenant le même modèle avec des corrélations inter-segments, atteignent un Q_3 de 69.2%, toujours sur séquences seules.

Chu et al [42] introduisent l'information des profils PSI-BLAST sous forme de distributions multinomiales, et intègrent des corrélations longues distances (formation de feuillets β) dans leur HSMM. Le score Q_3 atteint est de 72.8%. Les auteurs précisent qu'ils n'ont pas observé d'amélioration significative de la prédiction grâce à la prise en compte des corrélations longues distances. L'amélioration des résultats de Schmidler et al semble être

principalement due à l'utilisation des profils.

3.3 Conclusion

Ce chapitre a situé le cadre méthodologique des HMM et les principales stratégies de prédiction de structure secondaire utilisant les HMM. Les stratégies sont diverses : modélisation *automatique* des structures secondaires, construction de modèles spécifiques de classes, constructions de modèles à partir de librairies de fragments, prise en compte de fenêtre glissante dans la structure secondaire (revenant, à quelques dépendances près, à augmenter la mémoire du processus caché) ou de la taille des segments (HSMM).

Dans le chapitre suivant, nous présentons plusieurs approches de construction de HMM pour prédire la structure locale des protéines. Nous proposons des modèles pour prédire la structure secondaire. L'addition de descripteurs de la structure du coil est également envisagée.

Chapitre 4

Choix de modèles pour la prédiction de structure locale des protéines

Le principal objectif de cette thèse a été de mettre en place une méthode de prédiction de la structure locale des protéines d'après leurs séquences. Pour cela, nous utilisons les modèles de chaînes de Markov cachées (HMM pour *Hidden Markov Models*). En plus de disposer d'un cadre théorique solide (voir chapitre 3), les HMM permettent une modélisation explicite des données : les hypothèses posées sur le modèle sont visibles. Ils permettent donc de spécifier un modèle qui incorpore les informations biologiques disponibles sur les séquences à modéliser. Une approche alternative consiste à travailler avec des modèles peu contraints. L'apprentissage d'un modèle relativement souple permet de mettre en lumière des caractéristiques -éventuellement inattendues- des données.

Le principal obstacle à l'utilisation des HMM est l'absence de méthodes satisfaisantes pour choisir automatiquement la meilleure topologie de modèles complexes. Dans le cas de profils HMM, des approches ont été proposées pour optimiser le nombre d'états cachés. Krogh et al [109] proposent une heuristique dite de *chirurgie du modèle* qui consiste à supprimer des états *match* trop peu visités. A l'inverse, des états *match* sont ajoutés si les séquences empruntent très souvent les états *delete* lors de l'estimation des paramètres. Les profils HMM sont des HMM d'un type particulier : les états fonctionnent par triplet et la progression dans le graphe est unidirectionnelle. De la même manière, les algorithmes génétiques ont été appliqués à la recherche de topologie de HMM pour les signaux de séquences dans l'ADN [201]. Les modèles générés ont alors une taille de l'ordre d'une dizaine d'états cachés. Nicolas et al [134] ont proposé un algorithme de Monte Carlo par chaîne de Markov à saut réversible (*reversible jump MCMC*) qui permet de choisir la taille du modèle et l'ordre des chaînes de Markov pour des modèles HSMM appliqués à la recherche de promoteurs bactériens.

Cependant, dans le cas des modèles de topologies plus complexes, le nombre d'états cachés, l'ordre du processus observé, ainsi que les contraintes de départ sur les transitions restent du ressort exclusif du modélisateur. Je me suis donc intéressée au problème du choix de modèle, i.e., l'ordre des chaînes de Markov des observations et la topologie du modèle.

Dans un premier temps, des modèles 3 états utilisant des schémas variés de mémoire ont été utilisés. Ces modèles montrent très vite leurs limites. Ensuite, nous avons proposé un modèle M1M0 en utilisant des *a priori* biologiques sur l'architecture interne des structures secondaires permettant d'obtenir un score Q_3 de 67.7%. Enfin, des critères de

performances et des critères statistiques ont été utilisés pour choisir au mieux le nombre d'états cachés dans des modèles M1M0 construits sur des *a priori* de modélisation assez faibles, amenant un score Q_3 de 68%. Enfin la prédiction des zones d'angles à été abordée avec des modèles assez pauvres en modélisation, mais qui fournissent des résultats satisfaisants : la prédiction des 3 principales zones d'angles dièdres est correcte pour 72.7% des résidus.

4.1 Matériel et méthodes

4.1.1 Données

Les modèles sont entraînés et testés sur un ensemble de 2530 domaines structuraux de la banque de données ASTRAL 1.65 [31]. Seules les structures obtenues par cristallographie ayant un facteur de résolution inférieur à 2.25 Åsont retenues. C'est un jeu de données non-redondant en séquence : l'identité de séquence maximale entre paires de séquences est de 25%.

Les structures secondaires sont assignées, lorsque ce n'est pas précisé, par notre méthode KAKSI. Les assignations fournies par STRIDE [69] et DSSP [91] ont également été utilisées. Les assignations fournies par STRIDE et DSSP sont converties en 3 classes comme suit : DSSP, STRIDE : (H, G, I) = hélice, (E, b) = brin, autres (S, T, espace) = coil. Notre jeu de données correspond à 489743 résidus de structure secondaire définie (en excluant les résidus dont les coordonnées sont manquantes).

Les modèles sont entraînés et testés grâce à une procédure de **validation croisée** sur 2024 structures sélectionnées aléatoirement. Une validation croisée à 4 sous-ensembles est effectuée : trois quarts des structures servent à l'estimation des paramètres et un quart sert à tester les modèles. Les quatre partitions de l'ensemble de validation croisée contiennent respectivement 94790, 101521, 99796, 99031 résidus.

Les 506 structures restantes constituent l'**ensemble de test indépendant**. Cet ensemble de test n'est jamais utilisé pour estimer les paramètres. Il permet de contrôler que les modèles ne sont pas biaisés vers l'ensemble de validation croisée. Il correspond à 94605 résidus de structure secondaire définie.

Remarque : idéalement, un ensemble de test comme celui-ci ne devrait être utilisé qu'une seule fois, puis laissé de côté définitivement. Le nombre de structures contenues

dans la PDB ne permet pas d'opérer ainsi. La procédure utilisée ici (validation croisée + ensemble de test) est des plus rigoureuses, compte tenu des données disponibles.

Les contenus en structures secondaires sont similaires dans tous les sous-ensembles : environ 39% de résidus en hélice α, 24% en feuillet β et 37% en coil d'après les assignations de KAKSI. Ces taux sont respectivement de 38%, 22% et 40% avec STRIDE.

Une prédiction aléatoire des structures secondaires consistant à prédire au hasard 39% de résidus en hélice, 24% en feuillet et 37% en coil aboutirait à une prédiction correcte sur environ 35% des résidus.

4.1.2 Utilisation des HMM

Les modèles HMM sont identifiés et mis en oeuvre à l'aide du logiciel SHOW [135]. Ce logiciel offre une grande souplesse de modélisation, au travers d'une description textuelle du modèle construit par l'utilisateur.

4.1.3 Indices de prédiction

Les performances de prédiction sont évaluées au moyen de plusieurs indices.

Le score Q_3 est la proportion de résidus correctement prédits :

$$Q_3 = \frac{N_{ii}}{\sum_{i,j} N_{ij}}$$

avec N_{ij} le nombre de résidus observés dans la structure i et prédits dans la structure j. $\sum_{i,j} N_{ij}$ représente donc le nombre total de résidus soumis à la prédiction.

Le score SOV , défini par Zemla et al [208], est une mesure du recouvrement de segments de structures secondaires. Rappelons ici la définition donnée dans le chapitre 2. Pour la structure secondaire i, le recouvrement entre la structure secondaire réelle (s_1) et la structure secondaire prédite (s_2) est donnée par

$$SOV(i) = \frac{1}{N(i)} \sum_{s(i)} \frac{minov(s_1, s_2) + delta(s_1, s_2)}{maxov(s_1, s_2)} \times len(s_1)$$

avec $N(i)$ défini par :

$$N(i) = \sum_{S(i)} len(s_1) + \sum_{S'(i)} len(s_1).$$

Les sommes sur les $S(i)$ incluent toutes les paires de segments i ayant au moins un résidu de recouvrement. Les sommes sur les $S'(i)$ incluent les segments de structure secondaire de s_1 ne donnant pas lieu à des paires. $len(s_1)$ est le nombre de résidus du segment s_1, $minov(s_1, s_2)$ la longueur du recouvrement entre s_1 et s_2. $maxov(s_1, s_2)$ est l'étendue totale recouverte par les deux segments. $delta(s_1, s_2)$ est défini par :

$$min\left\{maxov(s_1,s_2) - minov(s_1,s_2); minov(s_1,s_2); int(\frac{len(s_1)}{2}); int(\frac{len(s_2)}{2})\right\},$$

avec $min\{x1; x2; x3; ...; xn\}$ le minimum de n entiers et $int(x)$ désignant la partie entière de x.

Le SOV global est donné par :

$$SOV = \frac{1}{N}\sum_{s(i)} \frac{minov(s_1,s_2) + delta(s_1,s_2)}{maxov(s_1,s_2)} \times len(s_1)$$

avec :

$$N = \sum_i N_i.$$

La sensibilité ou Q_{obs} pour une structure secondaire i est définie par :

$$Q_{obs}(i) = \frac{N_{ii}}{N_{obs_i}},$$

avec N_{ii} : le nombre de résidus *observés* dans la structure i et *prédits* dans la structure i, et N_{obs_i} : le nombre de résidus *observés* dans la structure i.

La spécificité ou Q_{pred} pour une structure secondaire i est définie par :

$$Q_{pred}(i) = \frac{N_{ii}}{N_{pred_i}},$$

avec N_{ii} : le nombre de résidus *observés* dans la structure i et *prédits* dans la structure i, et N_{obs_i} : le nombre de résidus *prédits* dans la structure i.

Le coefficient de corrélation de Matthew (CCM) [127] pour une structure secondaire i, tient compte à la fois de la sur-prédiction et de la sous-prédiction. Il est défini par :

$$C_i = \frac{VP_i \times VN_i - FP_i \times FN_i}{\sqrt{(VP_i + FN_i)(VP_i + FP_i)(VN_i + FN_i)(VN_i + FP_i)}}.$$

VP_i est le nombre de vrais positifs : résidus *observés* dans la classe i et *prédits* dans la classe i.

FP_i est le nombre de faux positifs : résidus *observés* dans une autre classe et *prédits* dans la structure i (ce qui correspond à une sur-prédiction).

VN_i est le nombre de vrais négatifs : résidus *observés* dans une autre classe et *prédits* dans une autre classe.

FN_i est le nombre de faux négatifs : résidus *observés* dans la structure i et *prédits* dans une autre structure (ce qui correspond à une sous-prédiction).

Le CCM est le coefficient de corrélation des indicatrices de la structure (indicatrices de vérité et indicatrices de prédiction). Pour une prédiction parfaite, le CCM est égal à 1, pour une prédiction aléatoire il est égal à 0.

4.2 Modèles à trois états cachés pour la prédiction de structure secondaire

Une modélisation très basique des structures secondaires consiste à modéliser chaque classe de structure secondaire par un unique état caché qui émet la séquence de la protéine. Dans ce cas, nous sommes en présence de données annotées. L'estimation des paramètres se fait donc par comptage : les probabilités du modèle sont estimées par les fréquences observées dans l'ensemble d'apprentissage (voir chapitre 3). La définition des structures secondaires selon notre méthode KAKSI, n'autorise pas l'apparition successive d'un résidu en hélice et d'un résidu en brin. Le graphe des états cachés du modèle obtenu sur la partition 1 de l'ensemble de validation croisée est montré dans la figure 4.1.

4.2.1 Modèles M1Mn

Dans ce cadre, le choix de modélisation porte sur l'ordre des chaînes de Markov des lois d'émission des états cachés. Nous avons testé des ordres allant de 0 à 4 pour les observations, l'ordre étant le même pour tous les états cachés. Le processus caché est toujours d'ordre 1. La prédiction des structures secondaires est réalisée avec les probabilités *a posteriori* calculées par l'algorithme forward/backward : en chaque site, l'état caché le plus probable est choisi pour la prédiction.

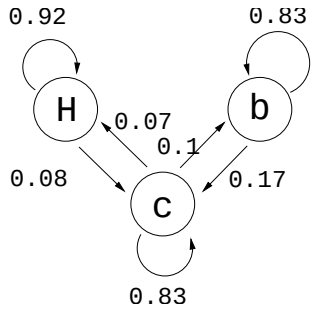

FIG. 4.1 – Graphe des états cachés d'un HMM 3 états pour la prédiction de structure secondaire. Les probabilités de transitions indiquées sur le graphe sont estimées sur la partition 1 de l'ensemble de validation croisée.

L'estimation de modèles d'ordres supérieurs à 1 nécessite l'introduction de pseudo-comptages. En effet, la quantité de données n'est pas suffisante pour prévenir l'estimation de probabilités nulles. Par exemple, pour un modèle d'ordre 3, l'estimation par comptage nécessite de compter les quadruplets d'acides aminés. Certains de ces comptages sont nuls, ce qui conduit à des probabilités d'émission nulles. Quand le modèle est testé sur des séquences qui contiennent des quadruplets non vus pendant l'apprentissage, l'algorithme forward/backward conduit à une vraisemblance nulle, rendant impossible la prédiction sur ces séquences.

La technique des pseudo-comptages consiste à prévenir l'estimation de probabilités nulles. Nous avons utilisé la méthode la plus simple, la règle de Laplace : ajouter 1 à tous les comptages. L'utilisation de pseudo-comptages plus élaborés, spécifiques de chaque mot, conduit à des résultats identiques.

Les scores Q_3 obtenus sur l'ensemble de validation croisée (moyennes sur les 4 partitions) et l'ensemble de test indépendant (moyennes sur les 4 modèles) sont rapportés dans le tableau 4.1.

Ces performances montrent que l'ordre 1 n'améliore pas significativement la prédiction par rapport à l'ordre 0. Dès l'ordre 2, on peut observer un phénomène de sur-apprentissage : la prédiction est bien meilleure sur les séquences d'apprentissage (Q_3=65.9%) que sur les séquences de test (Q_3=57.7%). Ce phénomène s'aggrave sur les ordres plus élevés. L'utilisation des pseudo-comptages permet de prévenir les paramètres nuls, mais pas d'estimer correctement les paramètres des modèles d'ordre élevé. Un modèle M1M4 est suffisant pour apprendre *par coeur* les données, puisqu'il peut prédire les structures secondaires avec un Q_3 de 98.2%. Par contre ce même modèle est très mauvais pour prédire

TAB. 4.1 – Scores Q_3 obtenus avec les HMM à 3 états cachés. * : utilisation de pseudo-comptages, voir texte.

Modèle	Nb Par [a]	Validation croisée		Test
		Apprentissage	Test	indépendant
M1M0	62	58.3%	58.3%	58.2%
M1M1	1 145	59.3%	58.7%	58.7%
M1M2	22 805	65.9%	57.5%*	58.1%*
M1M3	456 005	89.1%	47.6%*	47.5%*
M1M4	9 120 005	98.2%	41.7%*	41.7%*

[a]Nombre de paramètres indépendants

de nouvelles séquences (Q_3=41.7%, valeur guère supérieure à une prédiction alétoire).

Ce calcul préliminaire montre clairement que les modèles 3 états M1Mn sont d'un intérêt assez limité et qu'il convient d'utiliser d'autres schémas de prise en compte de la mémoire de la séquence. Ces approches sont présentées dans la suite de cette section.

4.2.2 Modèles MTD

Les matrices de Markov d'ordre élevés sont estimés à l'aide des modèles MTD, en utilisant l'estimation des paramètres décrite par Berchtold [18]. Rappelons ici la définition donnée dans le chapitre 3. Sous un modèle MTD, les influences de lettres précédentes sur la lettre au temps t sont combinées additivement :

$$\begin{aligned} P(X_t = x_t \mid X_{t-l} = x_{t-l}, ..., X_{t-1} = x_{t-1}) &= \sum_{g=1}^{l} \lambda_g P(X_t = x_t \mid X_{t-g} = x_{t-g}) \\ &= \sum_{g=1}^{l} \lambda_g q_{x_{t-g} x_t} \end{aligned}$$

avec la contrainte $\sum_g \lambda_g = 1$. Les $q_{x_{t-g} x_t}$ sont les éléments d'une matrice markovienne Q. Nous utilisons les modèles MTD dans lesquels la même matrice de Markov Q est utilisée pour les différents retards.

L'utilisation de matrices spécifiques pour chaque retard aboutit à un modèle sur-paramétré, ce qui ne permet pas d'utiliser la procédure d'estimation décrite par Berchtold (Sophie Lèbre, communication personnelle). Le tableau 4.2 récapitule les scores Q_3 obtenus par ces modèles avec l'algorithme forward/backward.

TAB. 4.2 – Scores Q_3 obtenus avec les HMM à 3 états cachés dont les lois d'émission sont estimées en utilisant les modèles MTD.

Modèle	Nb Par	Validation croisée		Test
		Jeux d'apprentissage	Jeux de test	indépendant
M1M2	1148	60.5%	59.9%	60.3%
M1M3	1151	59.9%	59.4%	59.8%

Le modèle MTD permet d'approximer un ordre 2, en évitant de manière significative le sur-apprentissage, mais les performances sont assez modestes : 60.3% de bonne prédiction. A l'ordre 3, le modèle MTD ne semble pas du tout adapté : les performances ne sont pas meilleures qu'à l'ordre 2.

Néanmoins, il est intéressant d'examiner les valeurs des paramètres des modèles MTD estimés sur nos données. En effet, les paramètres λ permettent de mesurer la force de l'association entre le retard (X_{t-l}) et le présent (X_t). Leurs valeurs sont rapportées dans le tableau 4.3, pour chaque structure secondaire.

TAB. 4.3 – Valeurs des paramètres λ des modèles MTD estimés sur la partition 1 de l'ensemble de validation croisée.

Ordre	Hélice	Brin	Coil
2	$\lambda_1 = 0.26$	$\lambda_1 = 0.68$	$\lambda_1 = 0.68$
	$\lambda_2 = 0.74$	$\lambda_2 = 0.32$	$\lambda_2 = 0.31$
3	$\lambda_1 = 0.21$	$\lambda_1 = 0.48$	$\lambda_1 = 0.54$
	$\lambda_2 = 0.59$	$\lambda_2 = 0.24$	$\lambda_2 = 0.22$
	$\lambda_3 = 0.20$	$\lambda_3 = 0.28$	$\lambda_3 = 0.24$

Pour les classes brin et coil, les λ_1 sont supérieurs aux autres λ. Par contre, pour les hélices, λ_2 est bien supérieur aux autres λ. Ces valeurs révèlent une caractéristique intéressante de la structure de dépendance entre sites dans les hélices : l'influence du résidu situé en $t-2$ est plus importante que celle du résidu situé en $t-1$ sur le résidu en t. Ce phénomène peut être vérifié par le calcul de l'information mutuelle.

L'information mutuelle entre deux variables X et Y prenant leurs valeurs dans un alphabet $\mathcal{A}$ est définie par :

$$IM(X,Y) = \sum_{x_i \in \mathcal{A}, y_j \in \mathcal{A}} P(X = x_i, Y = y_j) \times log_2 \frac{P(X = x_i, Y = y_j)}{P(X = x_i)P(Y = y_j)}.$$

Plus la corrélation entre X et Y est forte, plus cette quantité est importante. Si les variables X et Y sont indépendantes, elle est nulle.

Ici, l'information mutuelle est calculée entre la distribution des acides aminés d'un site, dans une structure secondaire particulière, et celles des sites voisins dans une fenêtre de taille 10. Les informations mutuelles calculées pour les hélices, les brins et les coils sont portées dans la figure 4.2.

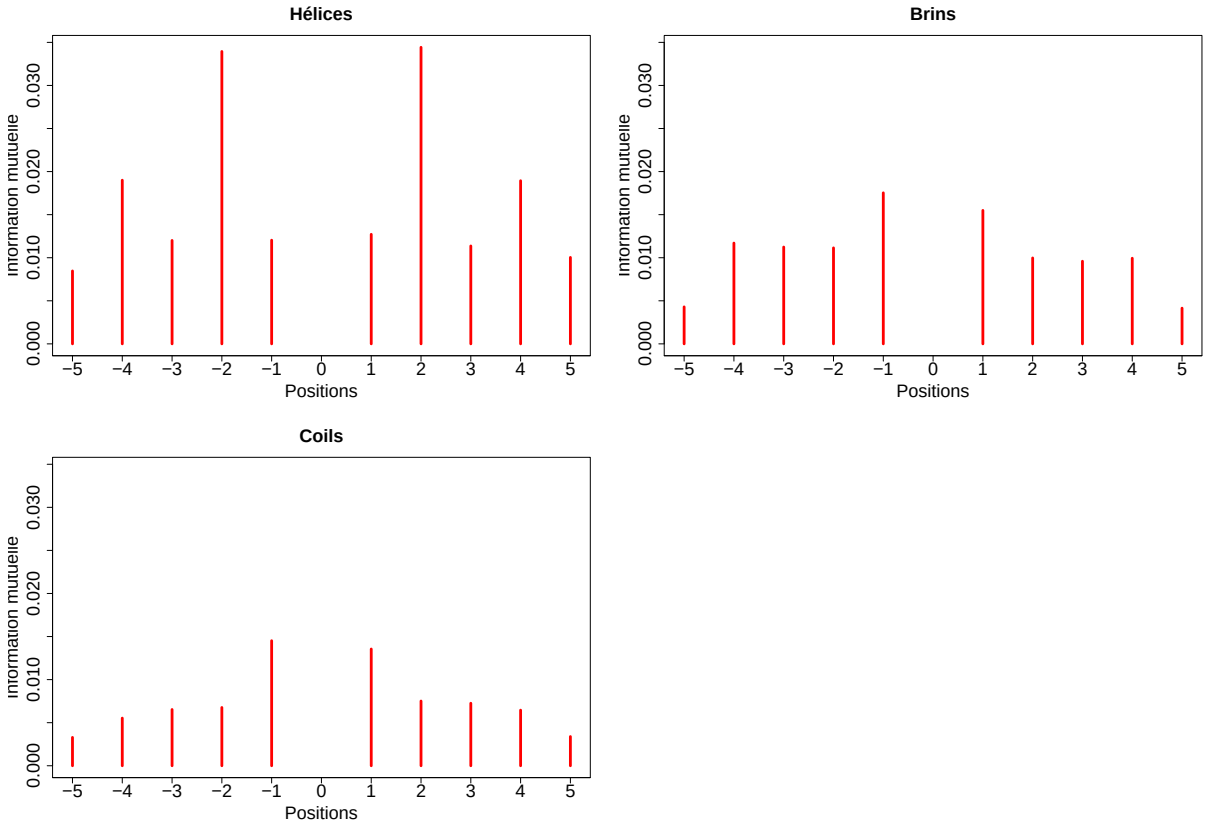

FIG. 4.2 – Information mutuelle entre sites voisins dans les hélices, brins et coils. L'information est calculée entre le site situé en 0, dans une structure secondaire donnée, et le site voisin sans distinction de conformation.

L'influence des résidus voisins sur les sites en brin et en coil décroît avec la distance. En revanche, dans les hélices, l'information mutuelle est plus importante entre les sites i et $i+2$ qu'entre i et $i+1$. Cette périodicité reste visible pour les sites en $i+4$.

L'explication biologique à cette structure de dépendance est la présence d'hélices amphiphiles [144, 170, 60, 92, 176]. Une hélice amphiphile est une hélice située à la surface d'une protéine. En conséquence, une face de l'hélice est en contact avec le reste de la protéine et l'autre baigne dans le solvant (l'eau). La face en contact avec la protéine porte

préférentiellement des résidus hydrophobes, et l'autre face, des résidus hydrophiles. La périodicité d'une hélice α étant de 3.6 résidus, la séquence montre une alternance de deux résidus hydrophiles et deux hydrophobes (parfois 2/1 au lieu de 2/2). Dans une hélice, les résidus situés en i et $i+2$ sont quasi-diamétralement opposés. Dans une hélice amphiphile, ils sont dans des environnements différents (eau/protéine), donc de natures opposées (hydrophile/hydrophobe). Par contre, le résidu situé en $i+1$ peut se trouver dans le même environnement que le résidu i ou non.

Cette caractéristique des séquences protéiques conduit à envisager l'apport des modèles *à trous*.

4.2.3 Modèles "à trous"

Dans cette modélisation, nous prenons en compte l'influence d'un seul résidu voisin, pas nécessairement adjacent. Par exemple, pour un modèle *à trous* d'ordre 2 : $P(X_n/X_1X_2...X_{n-2}X_{n-1}) = P(Xn/X_{n-2})$.

TAB. 4.4 – Scores Q_3 obtenus avec les HMM à 3 états cachés utilisant une chaîne de Markov *à trous* pour les lois d'émission.

Modèle	Nb Par	Validation croisée		Test
		Jeux d'apprentissage	Jeux de test	indépendant
M1sparseM2	1145	61.0%	60.4%	60.6%
M1sparseM3	1145	59.6%	58.9%	58.7%
M1sparseM4	1145	60.6%	60.1%	59.8%

Les performances de ces modèles avec l'algorithme forward/backward sont rapportées dans le tableau 4.4. Le meilleur taux de prédiction est obtenu avec un modèle *à trous* d'ordre 2 : 60.6%. Il est intéressant de noter que ce modèle est plus performant que le modèle M1M1 (58.7%). Les performances sont meilleures avec le modèle d'ordre 4 qu'avec le modèle d'ordre 3, ce qui est en accord avec les calculs d'information mutuelle. Les hélices étant la classe majoritaire, et montrant les plus fortes corrélations inter-sites, une meilleure prise en compte de la dépendance dans les hélices permet d'améliorer les performances.

Le fait que certaines dépendances peuvent *sauter* des sites amène à explorer l'apport d'une généralisation des chaînes de Markov à ordre variable, apparentée aux modèles à trous : les modèles de Markov parcimonieux (voir chapitre 3). Cette étude a été réalisée en

collaboration avec Pierre-Yves Bourguignon, à l'aide des programmes de la suite seq++ [130].

4.2.4 Modèles parcimonieux

Pour estimer les modèles parcimonieux, il est nécessaire de spécifier à l'avance des groupes d'acides aminés pour restreindre les possibilités de fusion dans l'arbre de suffixes. En effet, l'algorithme utilisé nécessite d'énumérer toutes les configurations possibles sous un noeud : toutes les fusions de contextes possibles, des 1-uplets aux n-uplets. Pour un alphabet de taille $\mathcal{X}$, le nombre de fusions possibles est de $2^{\mathcal{X}} - 1$ [25], ce qui devient problématique avec un alphabet de taille 20. En pratique, dans notre cas, le nombre de groupes ne doit pas dépasser 9. Dans cette étude, les mêmes groupes sont utilisés pour les trois classes de structure secondaire, mais rien n'empêche de proposer des groupements spécifiques pour chaque classe.

Premier groupement Une première estimation est menée en spécifiant les groupes suivants : (V, L, I), (A, G), (M), (P, F), (D, E), (H, K, R), (N, Q, C), (S, T) et (W, Y).

Les performances obtenues avec les lois d'émission estimées par modèle parcimonieux sont :

- validation croisée, apprentissage : $Q_3 = 60.6\%$,
- validation croisée, test : $Q_3 = 60.4\%$,
- test indépendant : $Q_3 = 60.7\%$.

Ces performances, limitées dans l'absolu, sont comparables aux performances du modèle *à trous* M1M2. Cette modélisation évite le phénomène de sur-apprentissage : les performances sont équivalentes sur les données d'apprentissage et les données de test.

Si ces performances s'avèrent relativement modestes, il est néanmoins intéressant d'examiner les arbres de suffixes correspondant à cette modélisation, car ils mettent en lumière la structure de dépendance dans chaque classe. Les arbres de suffixes obtenus pour les trois classes de structure secondaire sont montrés dans la figure 4.3.

L'arbre des hélices est le seul dans lequel il y a plus de noeuds au niveau -2 qu'au niveau -1. Cette structure est en accord avec le fait que la position -2 a plus d'influence sur le présent que la position -1, comme le montrent les calculs d'information mutuelle. L'arbre des brins a 2 noeuds à chaque niveau, les fusions obtenues montre clairement une

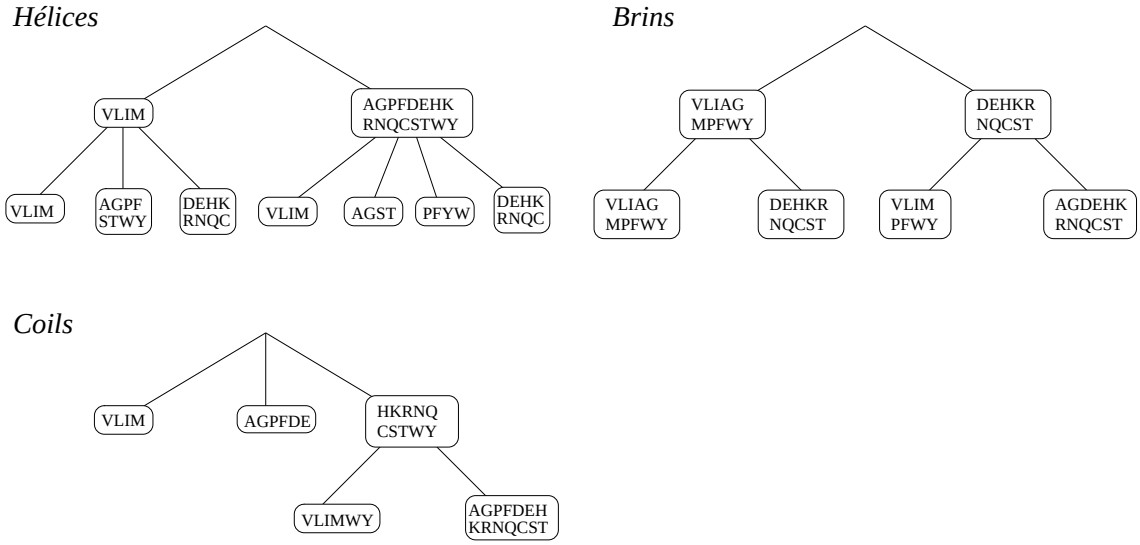

FIG. 4.3 – Arbres de suffixes obtenus par estimation de modèles parcimonieux sur les 3 structures secondaires, avec les groupes (V, L, I), (A, G), (M), (P, F), (D, E), (H, K, R), (N, Q, C), (S, T) et (W, Y). Ces arbres sont quasiment similaires sur les 4 partitions de l'ensemble de validation croisée.

séparation des résidus polaires et hydrophobes. L'arbre des coils est le seul à avoir été *élagué* en-dessous du niveau -1 : la mémoire de la séquence est *tronquée* au delà des contextes (VLIM) et (AGPFDE). Il y a, au total, 14 feuilles dans les 3 arbres. Le nombre de paramètres indépendants du modèle est donc :

$$5 + 14 \times 19 = 271.$$

Deuxième groupement Une deuxième estimation a été menée en autorisant les groupes suivants : (V, L, I), (A, G), (M, C), (P), (D, E), (H, K, R), (N, Q), (S, T) et (F, W, Y). La cystéine a été groupée avec la méthionine, acide aminé hydrophobe. La proline constitue un groupe à elle seule. La phénylalanine est groupée avec les aromatiques.

Les performances obtenues avec les lois d'émission estimées par modèle parcimonieux utilisant ces groupes sont très proches des résultats obtenues avec la première proposition :

- validation croisée, apprentissage : $Q_3 = 60.9\%$,
- validation croisée, test : $Q_3 = 60.7\%$,
- test indépendant : $Q_3 = 61.1\%$.

Les arbres de suffixes correspondants sont illustrés dans la figure 4.4.

Dans l'arbre des hélices, la proline a été isolée au niveau -1. L'arbre des coils est moins

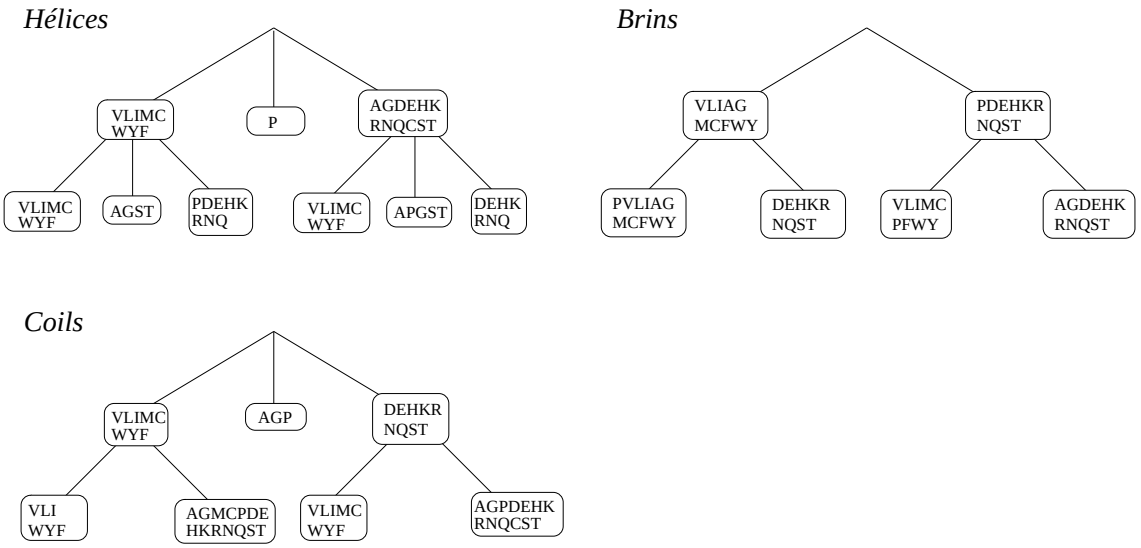

FIG. 4.4 – Arbres de suffixes obtenus par estimation de modèles parcimonieux sur les 3 structures secondaires avec les groupes :(V, L, I), (A, G), (M, C), (P), (D, E), (H, K, R), (N, Q), (S, T) et (F, W, Y).

élagué. Ce modèle à $5 + 16 \times 19$ =309 paramètres indépendants.

Ces résultats montrent que les dépendances entre sites dans les séquences de protéines sont assez complexes, et spécifiques de chaque classe de structure secondaire. Il serait souhaitable d'estimer dans un même temps les groupes d'acides aminés et les modèles parcimonieux, ce qui nécessite une approche par échantillonnage par Monte-Carlo, (développement en cours par Bourguignon et al).

4.2.5 Conclusion

Ces variations sur les modèles 3 à états montrent combien il est difficile d'utiliser des ordres élevés pour les lois d'émission des acides aminés, même en utilisant une prise en compte fine de la structure de dépendance. La modélisation a donc été orientée vers des modèles M1M0 ayant un plus grand nombre d'états cachés.

4.3 Proposition de modèle M1M0 construit avec des *a priori* biologiques

L'ordre du processus observé étant difficile à augmenter, nous proposons un modèle qui tient compte des connaissances *a priori* sur l'organisation des structures secondaires, revenant à établir un modèle M1M0 avec une topologie particulière. Topologie signifie ici : le nombre d'états cachés, les transitions entre ces états, éventuellement des contraintes sur la *nature* de ces états cachés, c'est à dire des contraintes sur les lois d'émission.

En s'inspirant des travaux sur la détection de gènes [134], nous essayons de proposer un modèle M1M0 de taille raisonnable dont la topologie rende compte de l'organisation intrinsèque des structures secondaires. Ce modèle sera utilisé comme point de départ pour l'estimation des paramètres par l'algorithme EM.

4.3.1 Modèle d'hélice α

Un motif bien caractérisé dans les hélices α est le motif amphiphile. Il est constitué d'une alternance de deux résidus polaires suivis de deux résidus apolaires. En utilisant une partition des acides aminés en deux classes : A,V,L,I,F,M,W,C=hydrophobes (h), S,T,Y,N,Q,H,P,D,E,K,R=polaires (p), les motifs hhpphh ou pphhpp sont retrouvés dans 24% des hélices de l'ensemble de validation croisée. Les motifs plus courts hhpp et pphh, correspondant à un tour complet d'hélice, sont retrouvés dans 69% des hélices. Les motifs amphiphiles sont donc particulièrement fréquents. Il est possible de tenir compte de ce motif particulier avec un modèle cyclique reproduisant l'alternance hydrophiles/hydrophobes. La figure 4.5 représente le graphe des états cachés proposé pour tenir compte de la nature amphiphile des hélices α.

Remarque : l'étude des mots exceptionnels (voir section suivante) dans les hélices révèle que les motifs correspondant aux hélices amphiphiles sont en réalité fortement surreprésentés.

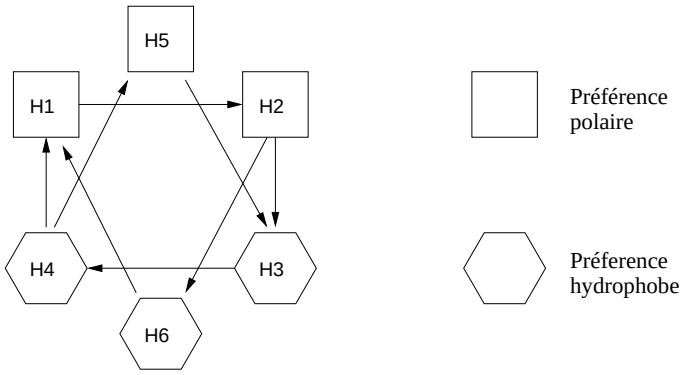

FIG. 4.5 – HMM représentant les hélices amphiphiles. Les états hydrophobes favorisent l'apparition des résidus A, V, L, I, F, P et M. Les états polaires favorisent l'apparition des résidus S, T, N, Q, H, D, E, K et R. La périodicité d'une hélice α est de 3.6 résidus. Ceci se traduit par l'apparition occasionnelle d'un seul résidu polaire (ou apolaire) au lieu de deux. Les états H5 et H6 permettent de tenir compte de cette caractéristique.

4.3.2 Modèle de brin β

Les brins β situés à la surface des protéines, à l'interface du solvant et du coeur de la protéine, présentent une alternance de résidus polaires et apolaires [120], [140, 125]. L'alternance polaire/apolaire peut être prise en compte par un modèle à deux états cachés, alternant un état hydrophobe et un polaire. Cette caractéristique des brins β est un motif moins fort que le motif amphiphile des hélices α [199].

Pour compléter ce modèle, une approche statistique fondée sur les mots exceptionnels [169, 168] est mise en œuvre.

Etude des mots exceptionnels dans les brins

Un mot est sur-représenté (respectivement sous-représenté) si sa fréquence observée est significativement supérieure (respectivement inférieure) à sa fréquence attendue sous un modèle Markovien donné. Les mots exceptionnels sont extraits avec le logiciel R'MES [1] [28, 169].

Les motifs recherchés sont, à la fois :

– exceptionnellement fréquents. Dans nos modèles, la séquence protéique est une chaîne de Markov d'ordre zéro conditionnellement à la chaîne cachée. Les mots

[1] http ://www-mig.jouy.inra.fr/ssb/rmes/

sur-représentés par rapport à un modèle d'ordre zéro sont donc intéressants car ils ne sont pas pris en compte par le modèle.

– fréquents dans l'absolu : il n'est pas question de fournir un effort de modélisation pour des motifs rares.
– spécifiques des brins β, donc non sur-représentés dans les hélices et le coil.

Certains mots très fréquents qui ne sont pas sur-représentés sont néanmoins retenus car ils sont sous-représentés dans les hélices. L'étude n'a pas permis de caractériser de mots sur-représentés dans le coil.

Les acides aminés sont divisés en deux groupes comme précédemment et la glycine est intégrée dans le groupe des hydrophobes. Les séquences des brins β et des hélices α de l'ensemble de validation croisée sont analysées séparément par R'MES. Le tableau 4.5 présente les motifs extraits des brins avec R'MES. L'exceptionnalité est évaluée par le calcul de e-value de R'MES et l'abondance relative est donnée par le classement des mots selon leur fréquence.

TAB. 4.5 – Motifs caractéristiques des brins β

Motif	Fréquence dans les brins	Fréquence dans les hélices
hphp	sur-représenté et fréquent	sous-représenté et peu fréquent
phph	sur-représenté et fréquent	sous-représenté et peu fréquent
pphhh	sur-représenté et très fréquent	sous-représenté et peu fréquent
pphph	sur-représenté et très fréquent	sous-représenté et peu fréquent
hhhhp	pas sur-représenté mais très fréquent	sous-représenté et peu fréquent
phhhhp	pas sur-représenté mais très fréquent	sous-représenté

Le modèle proposé pour tenir compte des motifs caractéristiques des brins β est représenté dans la figure 4.6.

Les mots hphp et phph sont pris en compte par l'alternance entre les états b1 et b2. La transition de l'état b4 sur lui-même favorise la répétition de résidus hydrophobes présents dans les mots pphhh, hhhhp, phhhhp. Ces répétitions de résidus hydrophobes correspondent à des brins enfouis dans le coeur des protéines [125]. La transition de l'état b2 vers b3 favorise l'apparition de deux résidus polaires, comme dans les motifs pphhh et pphph.

La recherche de mots exceptionnels dans le coil n'a pas permis d'extraire de motifs

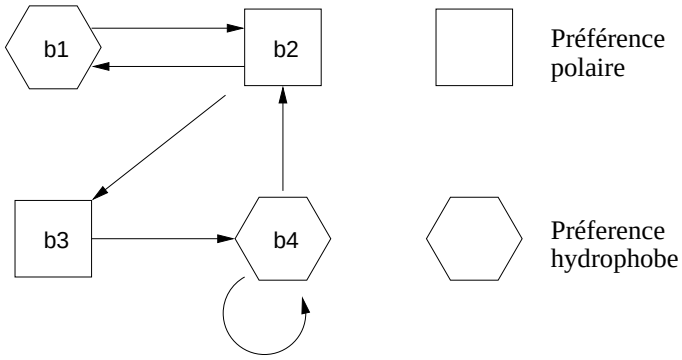

FIG. 4.6 – HMM représentant les brins β. Les états hydrophobes favorisent l'apparition des résidus A, V, L, I, F, P, M et G. Les états polaires favorisent l'apparition des résidus S, T, N, Q, H, D, E, K et R.

sur-représentés. L'analyse des mots sur-représentés dans cette classe, qui est une définition par défaut, requiert probablement une pré-classification des séquences.

4.3.3 Proposition d'un modèle complet pour les structures secondaires

Les modèles proposés pour les hélices α et les brins β sont fusionnés pour former le modèle complet présenté dans la figure 4.7.

Les modèles proposés pour les hélices et les brins (figures 4.6 et 4.5) tiennent compte des motifs caractéristiques identifiés dans les deux classes respectives. Les contraintes de modélisation sont très fortes : la plupart des transitions entre états ne sont pas autorisées, c'est pourquoi nous les appelons modèles *structurés*. Ces modèles structurés ne peuvent pas, à eux seuls, représenter pleinement les motifs rencontrés dans les deux classes. Pour modéliser les hélices et des brins qui ne correspondent pas aux motifs caractéristiques, deux états dénommés *échappatoires*, H7 et b5, sont ajoutés au modèle. Ces états ne favorisent pas *a priori* certaines classes de résidus. Ils communiquent avec tous les états des sous-modèles structurés. Quatre états supplémentaires (H8, H9, b6 et b7) sont ajoutés aux extrémités des hélices et des brins. Ils modélisent les signaux de séquence spécifiques de début et de terminaison d'hélices et de brins, comme les extrémités (*capping*) d'hélice [8, 9]. La classe coil n'est pas modélisée dans notre modèle complet, à l'exception des états aux frontières des hélices et des brins.

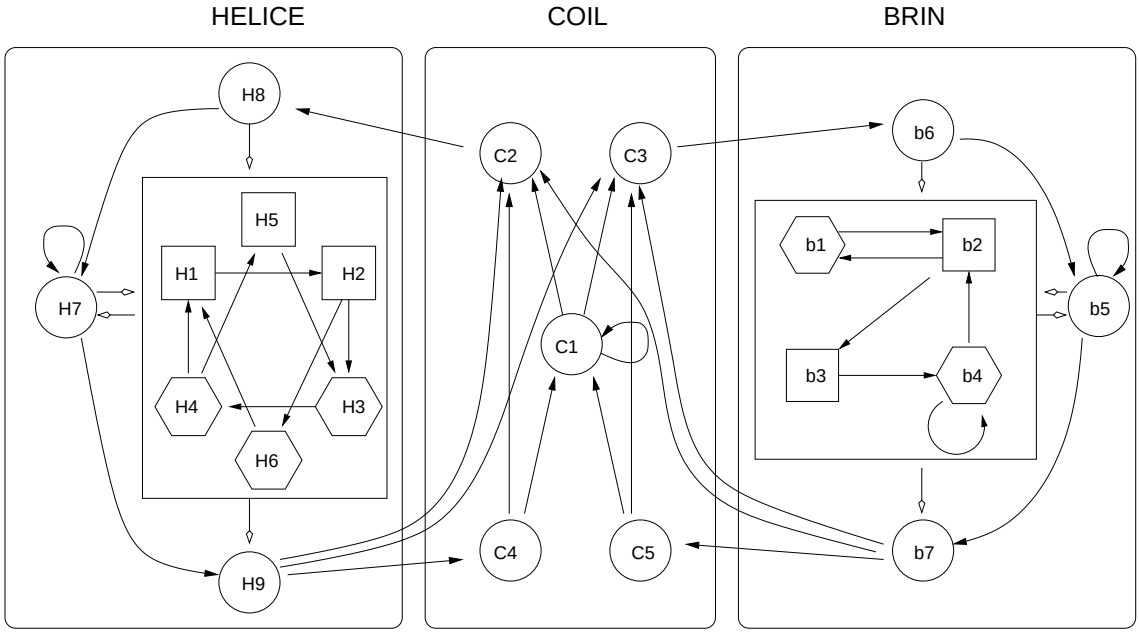

FIG. 4.7 – HMM complet des structures secondaires. Les flèches creuses indiquent une transition autorisée avec tous les états d'un sous-modèle.

4.3.4 Estimation des paramètres et utilisation du modèle

Valeurs initiales des paramètres

Le modèle ainsi construit sert de point de départ pour l'estimation des paramètres par EM. Les valeurs initiales des paramètres sont choisies comme suit :

- Les transitions autorisées sont égales à $\frac{1}{n}$, n étant le nombre d'états sortant.
- Les probabilités d'émission initiales sont dérivées des distributions obtenues sur le modèle M1M0 à 3 états cachés. Dans les états à préférence marquée, elles sont manuellement modifiées pour favoriser l'apparition de résidus polaires et défavoriser l'apparition de résidus apolaires, et vice-versa. Aucun biais n'est introduit pour les états sans préférence marquée.

Notre modèle a 21 états cachés et 468 paramètres indépendants. Ces paramètres sont estimés en utilisant le formalisme de *class HMM* (voir chapitre 3). Le HMM est bidimensionnel : un état caché émet simultanément la séquence protéique (paramètre à estimer) et l'étiquette de structure secondaire (paramètre fixe). La dimension *étiquette* est retirée du modèle après l'estimation.

Prédiction des structures secondaires

La prédiction des structures secondaires est réalisée par les probabilités *a posteriori* obtenues avec l'algorithme forward-backward. Les probabilités des états représentant la même structure secondaire sont sommées. La structure prédite est celle qui obtient la plus forte probabilité. Les différents scores de prédiction obtenus par ce modèle sont récapitulés dans le tableau 4.6.

TAB. 4.6 – Performances du modèle à 21 états cachés. Les 3 valeurs rapportées sont, dans l'ordre : le score moyen obtenu sur les séquences d'apprentissage, le score moyen obtenu sur les séquences de test de l'ensemble de validation croisée et le score moyen obtenu sur les séquences de test.

	Prédiction par classe				Prédiction globale	
	Sensibilité	Spécificité	CCM	Score SOV	Score Q_3	Score SOV
Hélice	74.7%	67.5%	0.55	70.1%	65.4%	60.8%
	74.1%	67.3%	0.55	69.3%	65.2%	60.5%
	75.7%	67.4%	0.56	69.7%	65.7%	60.4%
Brin	44.9%	63.9%	0.44	53.6%		
	45.0%	63.7%	0.44	53.7%		
	44.2%	64.8%	0.44	52.6%		
Coil	68.0%	63.8%	0.47	56.45%		
	68.0%	63.6%	0.47	56.48%		
	68.3%	64.3%	0.48	56.56%		

La prédiction est sensiblement améliorée par rapport aux modèles à 3 états cachés, atteignant un score Q_3 de 65.7%. Notre modèle évite totalement le sur-apprentissage. Les indices de prédiction par classe montrent que la prédiction des hélices est nettement meilleure que celle des autres classes. Les brins β sont assez pauvrement prédits par notre modèle : la sensibilité de prédiction n'est que de 44%.

A des fins de comparaison, les structures secondaires des séquences de test sont prédites à l'aide de la méthode PSIPRED, basée sur des réseaux de neurones [90]. Cette méthode effectue la prédiction à partir des profils de séquences. Comme notre modèle, à ce stade, n'utilise pas les profils, PSIPRED est utilisé en mode uniséquence. Un profil artificiel, contenant uniquement la séquence à prédire, est fournie en entrée du réseau de neurones. De plus, cette méthode a été optimisée pour reproduire les assignations de DSSP, or dans cette étude, KAKSI est utilisée comme référence. Il est donc important de souligner que

PSIPRED n'est pas utilisé ici de manière optimale. L'influence de la méthode d'assignation est abordée dans la section 4.4.4

Les performances de PSIPRED, dans ces conditions un peu particulières, sont reportées dans le tableau 4.7.

TAB. 4.7 – Performances de PSIPRED, en mode uniséquence, sur les séquences de test.

	Prédiction par classe				Prédiction globale	
	Sensibilité	Spécificité	CCM	Score SOV	Score Q_3	Score SOV
Hélice	69.7%	71.3%	0.55	71.1%	66.0%	62.7%
Brin	56.1%	60.7%	0.48	62.9%		
Coil	68.3%	64.0%	0.48	55.6%		

Les performances globales de ces deux systèmes sont équivalentes, avec un léger avantage pour PSIPRED concernant le score SOV. Par contre, PSIPRED donne des résultats nettement des meilleurs que le modèle HMM sur les brins β. Pour les hélices, le modèle HMM est plus sensible mais moins spécifique que PSIPRED, aboutissant à des CCM égaux. Signalons toutefois que PSIPRED comporte un étape de filtrage des résultats de prédiction par un deuxième réseau de neurones. Les prédictions fournies par notre HMM ne sont pas filtrées. Une étape de régularisation des prédictions pourrait peut-être permettre d'améliorer un peu la qualité de prédiction.

4.4 Choix d'un modèle M1M0 parmi des modèles générés automatiquement, sans *a priori*

Le modèle que je propose, à partir d'*a priori* biologiques, a 21 états cachés et des contraintes très fortes sur les transitions entre états cachés. La majorité des caractéristiques de ce modèle résultent donc directement du choix du modélisateur.

Une autre approche s'est révélée plus efficace : l'utilisation de modèles nettement moins contraints, ce qui laisse une plus grande part de liberté au modèle pendant l'estimation des paramètres.

Dans cette section, l'apport de modèles dans lesquels le seul choix de modélisation est le nombre d'états cachés constituant chaque classe structurale est envisagée. Dans ce cadre, le problème s'oriente vers la sélection de modèles, au moyen de critères objectifs

(performances en prédiction) et statistiques (critère BIC et distances statistiques entre modèles). Cette exploration commence par des modèles *équilibrés* ayant le même nombre d'états cachés pour modéliser chaque structure secondaire. Ces expériences montrent que les critères de performance et les critères statistiques s'accordent à sélectionner des modèles de taille relativement modeste. Nous continuons donc en essayant d'accroître séparément chaque classe, ce qui permet de sélectionner des plages de taille à tester pour les différentes classes. Le modèle optimal est sélectionné au moyen d'un critère statistique.

4.4.1 Estimation des paramètres et utilisation des modèles

Les lois d'émission initiales des paramètres sont aléatoires. Pour prévenir les problèmes de maximum locaux, 10 points de départ différents sont testés. Le meilleur modèle est sélectionné sur la base de la meilleure vraisemblance au cours de l'estimation par l'algorithme EM. Seul ce modèle est mené au bout de l'estimation. Toutes les transitions sont autorisées au sein de chaque classe structurale, ainsi qu'entre les classes coil et α et coil et β. Elles sont initialement égales à $\frac{1}{n}$, n étant le nombre d'états sortant. L'estimation utilise le formalisme des *class HMM*, comme précédemment. Les prédictions sont réalisées avec les probabilités *a posteriori* de l'algorithme forward/backward, en sommant les probabilités des états cachés d'une même classe.

4.4.2 Modèles ayant le même nombre d'états cachés par classe

Dans un premier temps, le nombre d'états cachés est choisi égal pour toutes les classes structurales. Des modèles avec 1 à 75 états cachés par classe sont estimés avec l'algorithme EM.

Evolution des performances

La figure 4.8 montre l'évolution du score Q_3 et du score SOV avec la taille des modèles.

L'évolution des scores Q_3 et SOV montrent qu'un bon taux de prédiction est atteint pour des modèles ayant de 10 à 15 états par boite : un Q_3 de 68.3% et un SOV de 65% avec 15 états cachés. L'addition d'états supplémentaires améliore peu ces performances. Avec des modèles ayant plus de 25 états par classe, on observe un sur-apprentissage modéré : les performances continuent d'augmenter légèrement sur l'ensemble d'apprentissage

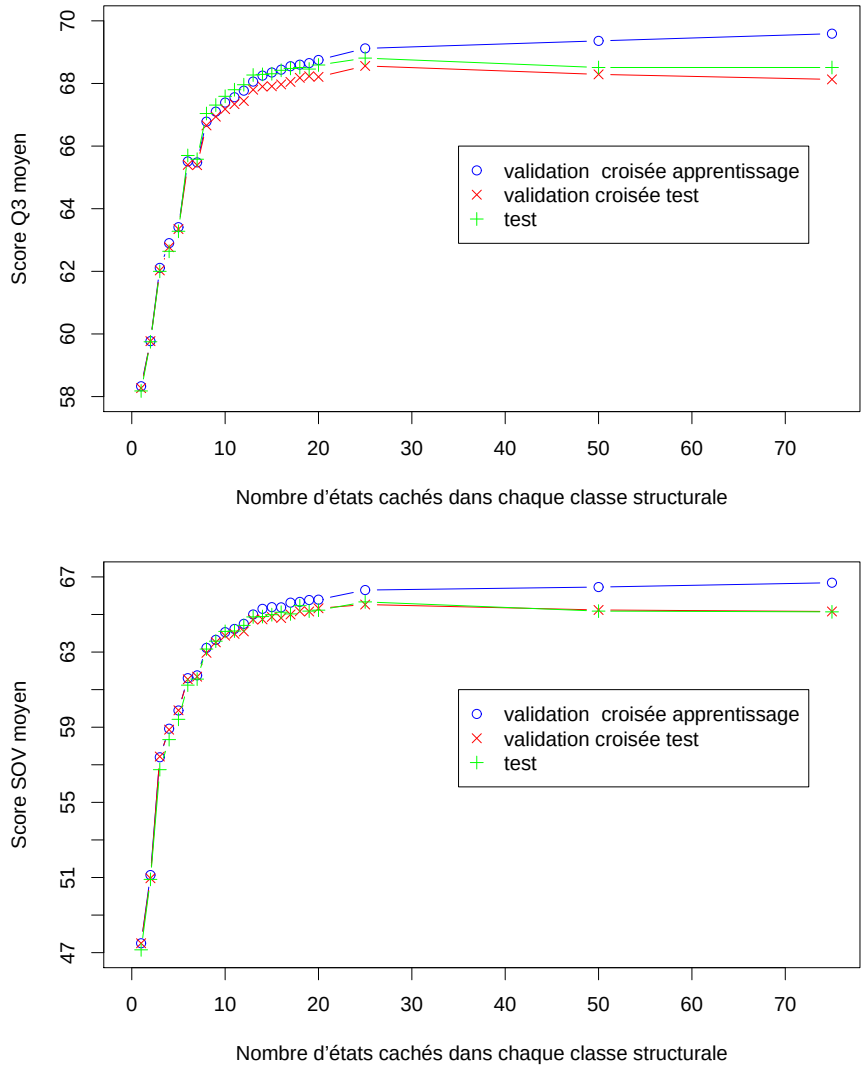

FIG. 4.8 – Evolution des scores de prédiction Q3 et SOV avec la taille des modèles

et décroissent sur les ensembles de test. Le Q_3 est ainsi de 69.6 % sur les ensembles d'apprentissage et de 68.1% sur les ensembles de test avec 75 états cachés par classe. Les modèles ayant plus de 25 états cachés par classe ont un nombre de paramètres trop élevé. Ils capturent des caractéristiques spécifiques de l'ensemble d'apprentissage et prédisent moins bien les séquences non vues pendant l'apprentissage (sur-apprentissage).

Pour s'assurer que le plateau de performance observé au-delà de 10 états cachés par classe n'est pas dû au problème d'optimum local de l'algorithme EM, les modèles ayant 10 à 25 états par boîte sont estimés avec 100 points de départ aléatoires. La figure 4.9

montre l'évolution du score Q_3 obtenu en prédiction avec ces modèles.

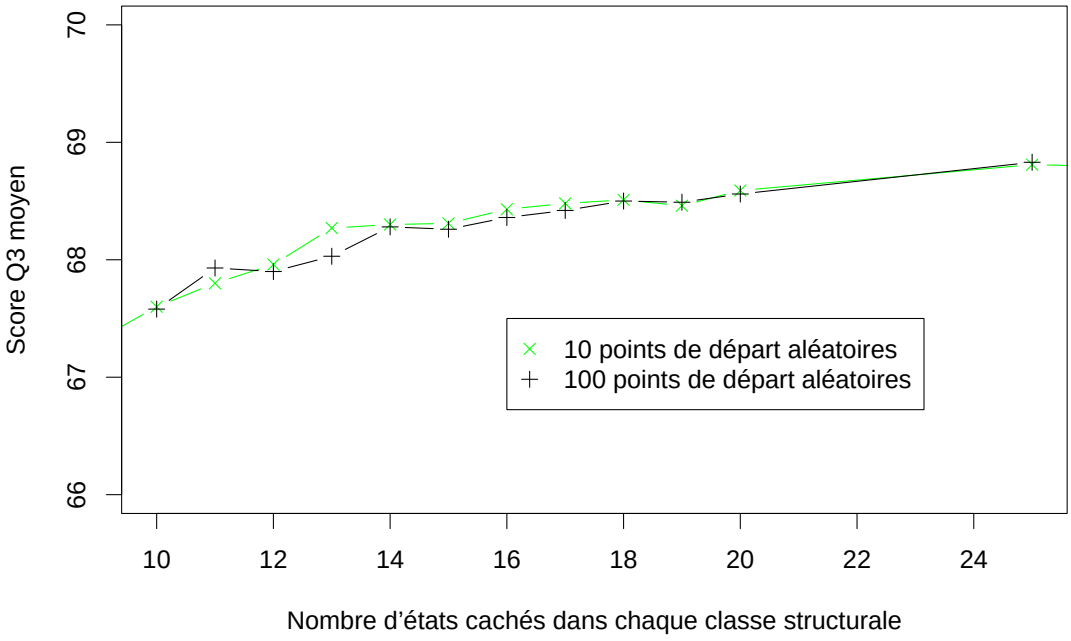

FIG. 4.9 – Evolution du score Q_3 sur l'ensemble de test indépendant

Les performances sont équivalentes quelque soit le nombre de points de départ testés, ce qui indique que 10 points de départ suffisent même pour des modèles de cette taille.

Critère statistique

L'évolution du taux de performance montre qu'un plateau est atteint pour des modèles de taille relativement modeste. Cependant il n'est pas aisé de choisir la taille optimale d'après ces résultats. L'estimation de ces modèles se situe dans un cadre semi-supervisé, car au moment de l'estimation, l'étiquette des états cachés est connue. Le critère BIC (Bayesian Information Criterion), utilisé dans les cas d'apprentissage non supervisé, peut être appliqué à notre étude. Le BIC [175] est un critère de vraisemblance pénalisé :

$$BIC = logL - 0.5 \times klog(N),$$

avec $logL$ la log-vraisemblance des données d'apprentissage sous le modèle à la fin de l'optimisation des paramètres, k, le nombre de paramètres indépendants du modèle et

N, la quantité de données utilisée pour l'estimation (nombre de résidus dans l'ensemble d'apprentissage). La vraisemblance est un terme qui augmente avec la taille du modèle car les modèles les plus gros représentent de mieux en mieux les données d'apprentissage. L'idée du critère BIC est de pénaliser la vraisemblance par un terme tenant compte de la taille du modèle et de la quantité de données effectivement disponibles pour estimer les paramètres. Le critère BIC permet ainsi de réaliser un compromis entre la représentation des données d'apprentissage et une taille raisonnable de modèle. Ce critère ne tient pas compte des performances prédictives des modèles.

Remarque Toutes les transitions initiales sont autorisées dans ces modèles. A la fin de l'optimisation, certaines transitions sont quasi-nulles. Le nombre de paramètres indépendants du modèle pour le calcul du BIC correspond au nombre de paramètres non nuls.

La figure 4.10 montre l'évolution du BIC sur les 4 sous-ensembles d'apprentissage avec la taille des modèles. Le maximum du BIC est obtenu pour des tailles de modèles de 13 à 14 états cachés par classe sur les différents sous-ensembles. Cette gamme de tailles de modèles est en bon accord avec les scores Q_3.

L'évolution des indices de performance et du critère BIC orientent le choix vers des modèles de taille modeste. Une dernière analyse intéressante consiste à comparer entre eux les différents modèles. Cette comparaison est effectuée grâce à un calcul de distances utilisant les séquences générées.

Distances entre modèles

La distance statistique entre modèles est calculée par la formule décrite par Rabiner [149]. La distance entre deux modèles M_1 et M_2 est donnée par :

$$D(M1, M2) = \frac{1}{T} |\log L(O^{(2)}|M_1) - \log L(O^{(2)}|M_2)|$$

$O^{(2)}$ désigne une séquence de longueur T *simulée* par le modèle M_2. $\log L(O^{(2)}|M_1)$ désigne la log-vraisemblance de cette séquence *sous* le modèle M_1, et $\log L(O^{(2)}|M_2)$ désigne la log-vraisemblance de cette séquence *sous* le modèle M_2 . Cette distance compare les lois de probabilités sur les séquences.

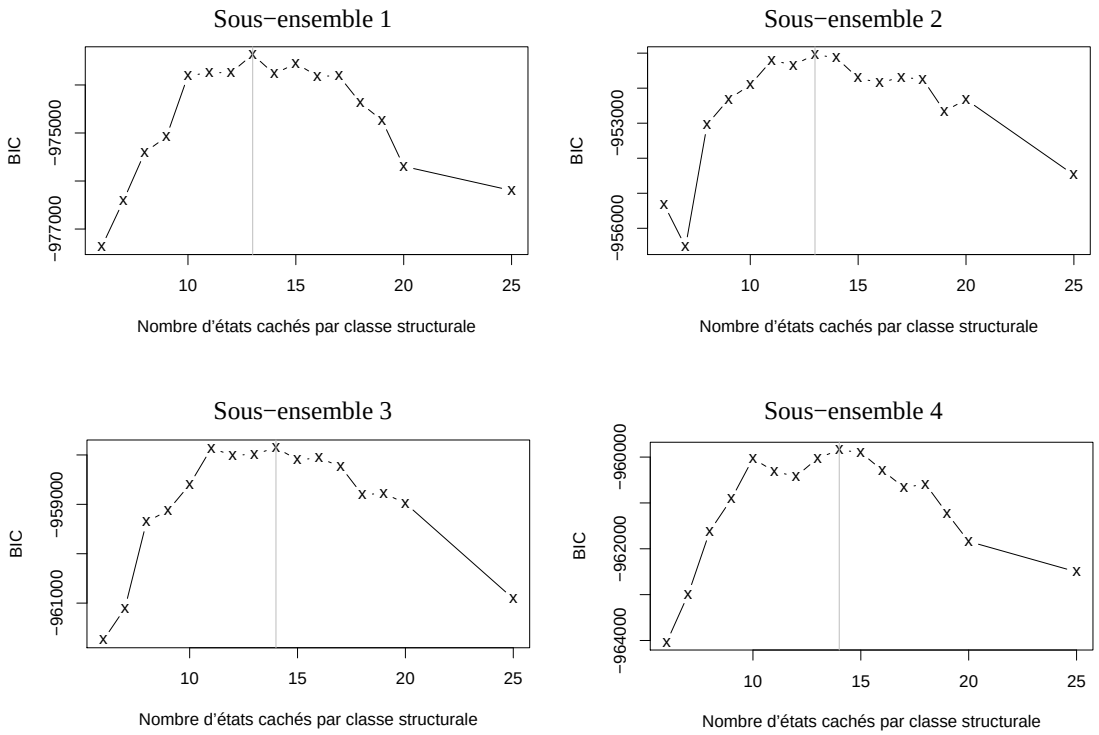

FIG. 4.10 – Critère BIC calculé sur les modèles de taille croissante. Le maximum du BIC est indiqué par une barre verticale.

Cette distance est rendue symétrique par :

$$D_s(M_1, M_2) = \frac{D(M1, M2) + D(M2, M1)}{2}$$

Ce calcul de distance ne tient compte ni des performances prédictives, ni de la taille des modèles. Il s'intéresse aux propriétés statistiques des séquences générées par les modèles.

Deux types de distances sont calculées entre les modèles :

- La distance intra-modèle concerne la comparaison de deux modèles ayant le même nombre d'états. 4 modèles étant estimés sur les 4 sous-ensembles d'apprentissage, la différence rapportée est la distance moyenne des 6 distances entre paires de modèles.
- La distance inter-modèles est la distance moyenne entre modèles n'ayant pas le même nombre d'états, estimés sur le même sous-ensemble d'apprentissage.

Toutes ces distances sont rapportées sur la figure 4.11. Comme attendu, les distances intra-modèles, figurant sur la diagonale, sont généralement faibles. Les modèles de même taille estimés sur des ensembles différents sont donc équivalents. Pour des modèles ayant 13 à 17 états par classe, les distances inter-modèles sont relativement faibles. Ces faibles

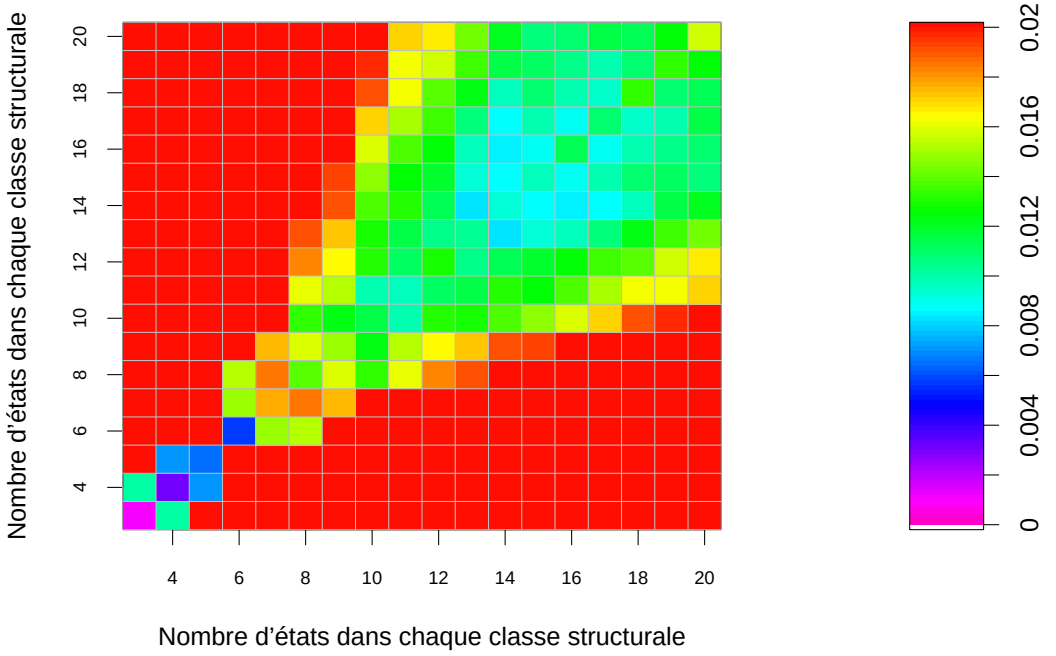

FIG. 4.11 – Distances statistiques symétriques entre modèles

distances indiquent que des modèles avec 13 à 17 états par classe génèrent des séquences statistiquement très proches et sont donc statistiquement équivalents.

Les performances prédictives, le BIC et les distances statistiques entre modèles suggèrent une taille de modèle d'environ 15 états cachés par classe pour obtenir des bonnes performances et avoir un nombre de paramètres raisonnable.

4.4.3 Modèles ayant un nombre d'états cachés différent dans chaque classe

L'étude précédente suggère des tailles relativement modestes. Cependant, l'hypothèse de classes structurales ayant le même nombre d'états cachés est une hypothèse forte.

Environ 30% des résidus sont impliqués dans les hélices α et 20% dans les feuillets β. La quantité de données disponible pour l'estimation des paramètres n'est pas la même pour les trois classes. Certaines classes vont *saturer* plus vite que d'autres. Dans le calcul du BIC, cet effet correspond au terme faisant intervenir le nombre de paramètres et la quantité de données.

Un deuxième effet s'ajoute à la répartition inégale des données : la complexité intrinsèque de chaque classe, en terme de distribution des acides aminés. Par exemple, une classe pourra être bien modélisée par un nombre limité de distributions d'acides aminés. Une autre, plus complexe, nécessitera peut-être la spécification d'un plus grand nombre d'états cachés. Cet effet se traduit dans le BIC par le terme de vraisemblance des données. La complexité des classes pourra être en contradiction avec l'effet de taille des classes (si la classe la moins abondante est la *plus complexe*), mais il est difficile d'avoir une idée de la complexité de chaque classe *a priori*.

Etude séparée de chaque classe

Pour estimer la gamme de taille à explorer pour chaque classe structurale, le nombre d'états cachés est augmenté dans chaque classe séparément. Les autres classes sont toujours modélisées par un seul état caché. La figure 4.12 rapporte l'évolution du score Q_3 obtenu avec ces modèles. L'accroissement du nombre d'états cachés modélisant les hélices α augmente significativement le score de prédiction globale. Ce résultat est attendu, puisque les hélices α sont plus abondantes que les brins β.

L'augmentation du nombre d'états cachés modélisant le coil a peu d'impact sur le Q_3, alors que cette classe englobe près de 40% des résidus. La classe coil est une définition par défaut, englobant tous les résidus non-α et non-β. Elle regroupe un ensemble de conformations locales aux caractéristiques probablement différentes, d'où ce faible accroissement de Q_3.

Le critère BIC a également été appliqué sur ces modèles. Les maximums du BIC sont obtenus pour 15 états cachés sur les modèles étudiant la classe hélice, 8 états cachés sur les modèles pour les brins et 9 états cachés pour les coils.

Choix de la combinaison optimale des tailles des sous-modèles

Rien ne garantit que le meilleur modèle ait 15 états cachés dans la boîte hélice, 8 dans la boîte brin et 9 dans la boîte coil. Ces tailles de sous-modèles ont été obtenues séparément, avec des modèles dans lesquels deux classes structurales sont modélisées par un seul état caché. Or, dans un modèle complet, les différentes classes sont connectées entre elles. Ces connexions entre classes sont probablement assez complexes, et se traduiront par la spécialisation de certains états cachés aux frontières de classes. Pour déterminer

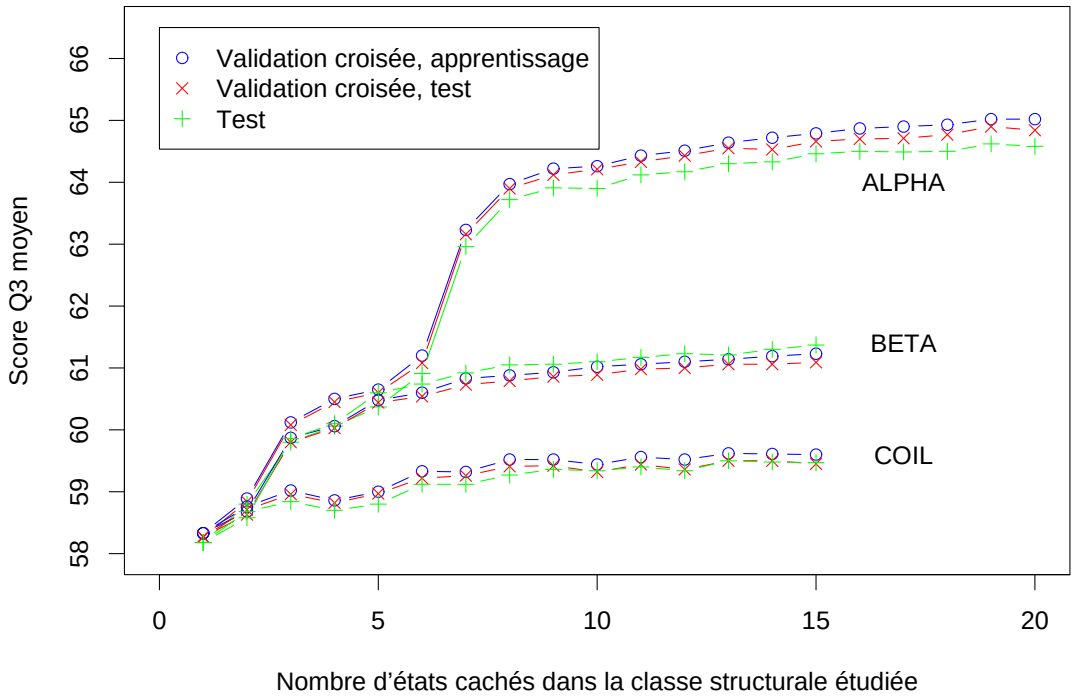

FIG. 4.12 – Evolution du score Q_3 quand le nombre d'états cachés augmente dans une seule classe structurale.

la combinaison optimale des nombre d'états cachés dans chaque classe, le critère BIC est calculé pour tous les modèles ayant 12 à 16 états cachés modélisant les hélices, 6 à 10 états cachés modélisant les brins et 5 à 13 états cachés modélisant le coil.

Le maximum du BIC est obtenu pour le modèle ayant 15 états cachés pour les hélices, 9 états cachés pour les brins et 12 états cachés pour le coil. La description du modèle optimal ainsi que ses performances en prédiction sont détaillées ci-dessous.

Description du modèle optimal

Les HMM permettent une modélisation explicite des données. Un modèle entraîné avec peu de contraintes *a priori*, comme c'est le cas ici, *apprend* la structure des données qui lui sont présentées pendant l'apprentissage.

Examen des transitions entre états cachés Lors de l'initialisation du modèle, toutes les transitions sont autorisées entre les états cachés modélisant les hélices et les états cachés modélisant le coil, de même qu'entre les états brin et les états coil. Après estimation des

paramètres, un certain nombre de ces transitions sont nulles. Le nombre de transitions initiales et finales entre les différentes classes structurales sont récapitulées dans le tableau 4.8.

TAB. 4.8 – Transitions dans le HMM final

Transitions	Nombre de transitions initiales	Nombre de transitions estimées non nulles	Ratio
Hélice vers Hélice	225	78	35%
Hélice vers coil	180	55	31%
Brin vers brin	81	45	56%
Brin vers coil	108	22	20%
Coil vers coil	144	88	62%
Coil vers hélice	180	31	17%
Coil vers brin	108	31	29%
Total	1026	350	34%

Ainsi, environ un tiers des transitions subsistent à l'intérieur de la boîte hélice, la moitié dans la boîte coil et deux tiers dans la boîte brin. Le sous-modèle des hélices est donc plus *structuré*, dans le sens où les transitions sont moins nombreuses, les chemins dans le graphe sont donc plus contraints.

Le modèle final, constitué de 36 états cachés a 350 transitions non nulles et, au total, 998 paramètres indépendants.

La figure 4.13 présente un aperçu graphique de la matrice de transition entre états cachés. Cette matrice de transition est très creuse : le nombre de connexions entre états cachés est limité. Cette représentation met en évidence des états cachés correspondant aux transitions entres classes structurales.

Le graphe des états cachés est présenté dans la figure 4.14. Par souci de clarté, seules les transitions dont les probabilités sont supérieures à 0.1 sont indiquées.

Ce modèle reflète l'organisation interne des séquences protéiques en structure secondaire, telle qu'elle a été apprise par le HMM. Les trois classes structurales montrent des architectures bien distinctes. L'architecture de la classe hélice montre une progression dans le graphe :

– entrée par les états 10, 9, 7 , 12, 6 ou 3,

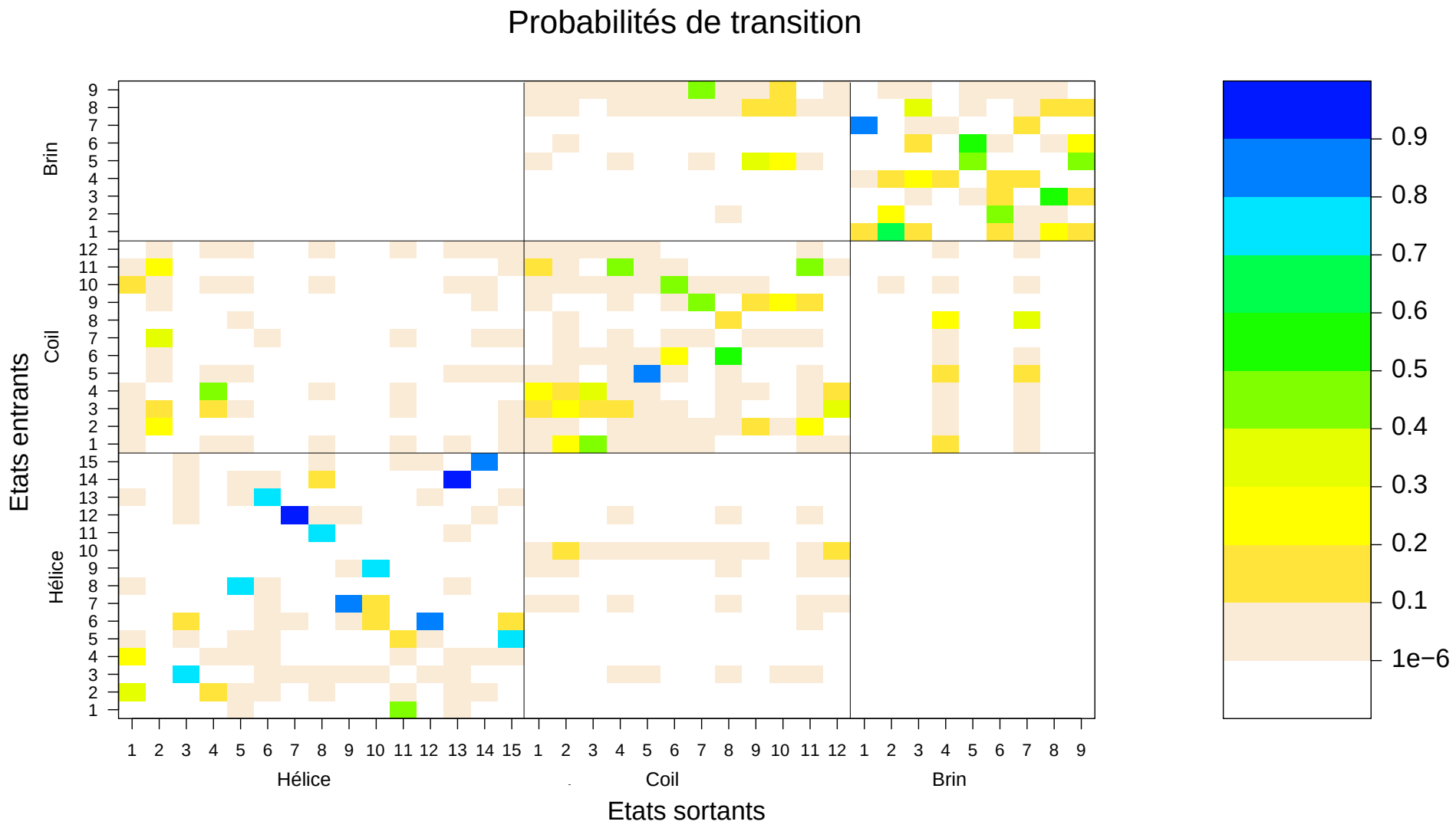

FIG. 4.13 – Matrice de transition entre états cachés du HMM optimal

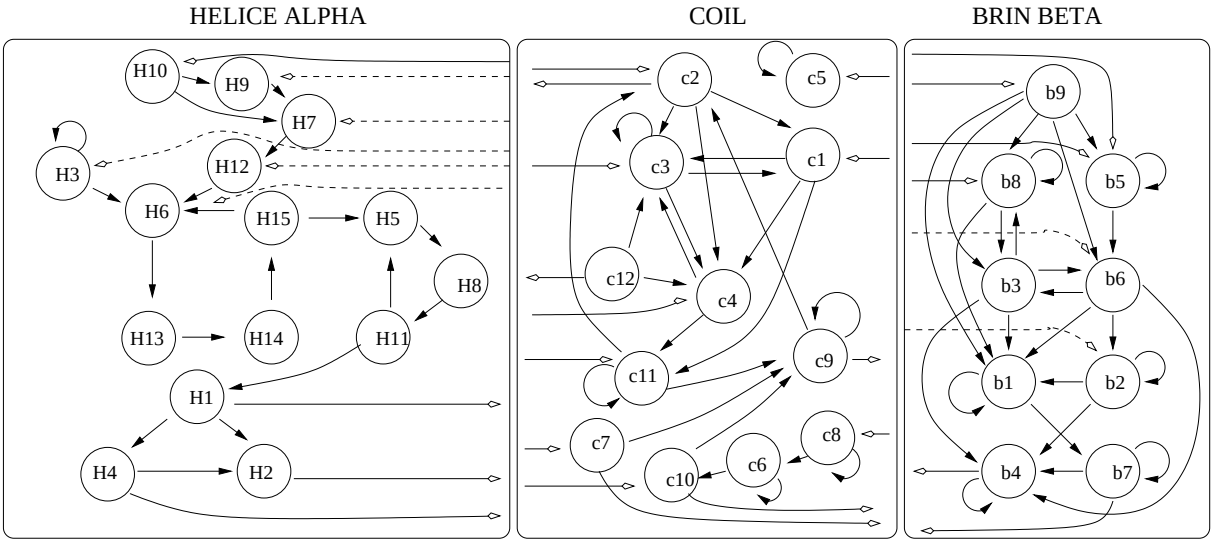

FIG. 4.14 – Topologie du HMM optimal. Les transitions de probabilités supérieures à 0.1 sont indiquées par un trait plein. Certaines transitions associées à des probabilités inférieures à 0.1 sont indiquées par des traits en pointillés : il s'agit des entrées dans les hélices et les brins. Les flèches creuses représentent des transitions entre classes structurales, non précisées pour la clarté du schéma.

- deux cycles : l'un à 4 états cachés (6, 13, 14 et 15), un autre à 3 états cachés (5, 8 et 11),
- sortie par les états 1, 2 ou 4.

La plupart des états cachés de l'hélice sont des états transitoires : les probabilités de rester dans un même état sont faibles. Dans le modèle que nous avions proposé sur des bases biologiques (4.3), nous avions distingué le début, le corps (cyclique) et la fin des hélices. Cependant, nous n'avions défini qu'un seul état de début et un seul état de fin.

La classe brin montre une organisation moins *structurée*, mais également directionnelle :

- entrée par les états 8, 9, 5, 2 ou 6,
- un corps de brin d'organisation relativement complexe (états 1, 2, 3, 5, 6 et 8) avec notamment deux cycles à deux états : 3/8, et 3/6,
- sortie par les états 4, 7 et 2.

La plupart des états cachés (6 sur les 9) ont des probabilités de transitions vers eux-même supérieures à 0.1. Ce modèle est bien plus complexe que le modèle précédent, mais comporte également une distinction début/corps/fin.

La classe coil révèle des états cachés spécifiques de frontières de brins et d'hélices. Les

états 2 et 12 sont des états pré-hélice, les états 2, 3, 4, 7, 10 et 11 sont des états post-hélice. Les états 1, 5 et 8 sont des états post-brin, et les états 7, 9 et 10, des états pré-brin. Un enchaînement de quatre états cachés permet une connexion de et vers les brins (états 8, 6, 10 et 9). Le coil était très peu modélisé dans le modèle à 21 états. La topologie observée ici est donc bien plus complexe notre précédente proposition.

Examen des lois d'émission des états cachés Les lois d'émission des états cachés sont illustrées dans la figure 4.15. Tous les états cachés montrent des préférences marquées en terme de distribution d'acides aminés.

Des études ont été publiées sur les spécificités de séquence dans les hélices [8, 113, 61]. Dans ces travaux, les auteurs comparent les fréquences des acides aminés à différentes positions dans les hélices à leurs fréquences globales. Il en ressort que la distribution des acides aminés dépend de la position dans l'hélice. Notre modèle montre une progression dans le graphe d'états, ce qui est en accord avec ces observations. Plusieurs états sont distingués pour les débuts et fins d'hélices : le modèle a donc identifié plusieurs signaux de séquences spécifiques. Les études publiées sur les extrémités d'hélices identifient les préférences d'acides aminés en *alignant* toutes les hélices d'une banque de structures : tous les premiers résidus d'hélices sont rassemblés pour calculer les fréquences en chaque position. Or notre modèle semble indiquer qu'il existe plusieurs *familles* de début d'hélice. La comparaison avec les préférences identifiées précédemment n'est donc pas immédiate, puisque celles-ci offrent une vue moyenne des préférences.

Kumar et Bansal ont identifié les acides aminés sur-représentés dans les débuts et fin d'hélices [113]. Dans cette étude, les hélices sont assignées par le programme DSSP, et sont éventuellement raccourcies sur des critères géométriques. Notre modèle a appris les structures secondaires définies par KAKSI, les hélices sont donc probablement un peu plus longues que si elles avaient été assignées par Kumar et Bansal (voir chapitre 2). Il faut donc tenir compte d'un éventuel décalage dans les préférences positionnelles. Dans notre modèle, l'état 10 montre une préférence pour l'aspartate, l'alanine, l'isoleucine et la valine. La préférence pour l'aspartate est répertoriée par Kumar et Bansal, aux positions 1, 2, -1, -2 et -3 des hélices (les indices négatifs désignent les positions précédant une hélice). Les préférences pour A, I et V ne sont pas retrouvées. Les préférences les plus marquées répertoriées par Kumar et Bansal au voisinage du début d'hélice portent sur la proline

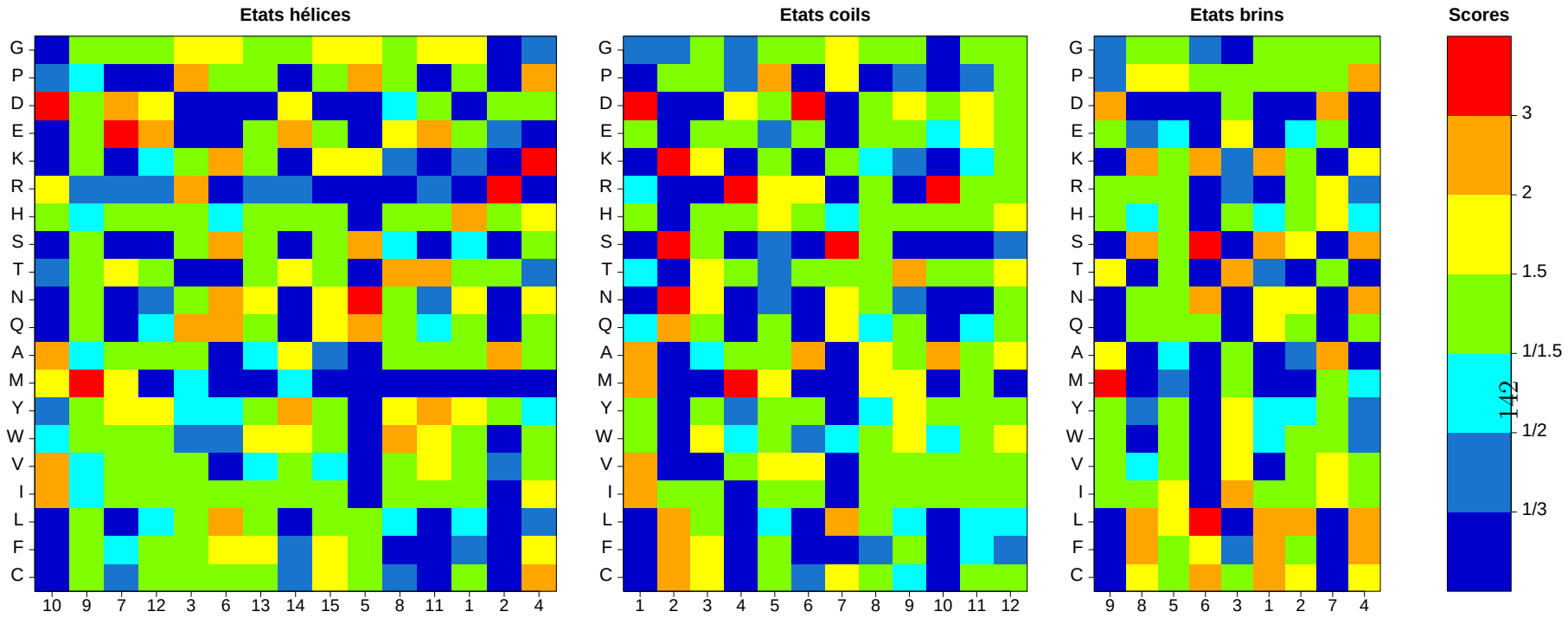

FIG. 4.15 – Lois d'émission des états cachés du HMM optimal. Les fréquences des acides aminés sont normalisées par rapport à leurs fréquences globales dans la banque de données : $Score(A, u) = \frac{P(A|u)}{f(A)}$, avec $P(A \mid u)$ la probabilité d'émission de A dans l'état caché u et $f(A)$ la fréquence globale de l'acide aminé A. Les états cachés des hélices et des brins ont été ordonnés selon la progression dans le graphe d'états (entrée, corps, sortie).

et les résidus polaires : aspartate, glutamate, glutamine, sérine, thréonine. Dans notre modèle, l'état 3 montre une préférence pour la proline. Les états 7 et 12 montrent une préférence pour le glutamate et l'aspartate. L'état 9 privilégie fortement la méthionine, qui est retrouvée préférentiellement en position +4 par Kumar et Bansal. Les résidus glycine et proline sont des interrupteurs d'hélices [9], et identifiées comme telles par Kumar et Bansal. Ils sont sur-représentés par les états H1 et H4 du modèle. La préférence pour la lysine de l'état 4 est identifiée au voisinage des fins d'hélice.

Engel et DeGrado ont montré une très nette périodicité dans la distribution des acides aminés dans le corps des hélices (au delà du cinquième résidu) : l'alternance de résidus de groupes physico-chimiques distincts, tous les 3 ou 4 résidus [61]. Notre modèle présente deux cycles a 3 et 4 états cachés, suggérant que le modèle a identifié une certaine périodicité. Cependant, il est assez difficile de dire, d'après les préférences de ces états cachés, s'ils traduisent la même alternance que celle identifiée par Engel et DeGrado.

Les préférences aux extrémités de brins β n'ont pas été caractérisées, dans la littérature. De manière générale, la spécificité de séquences dans les brins est moins étudiée. Les données sont moins nombreuses que pour les hélices, et les préférences semblent moins marquées. Les états de début de brin dans nos modèles montrent des préférence pour les résidus : M et D (état 9), K, S, L, F (état 8) et L, S, K, N, C (état 5). Les fins de brins sont marquées par deux distributions assez différentes : l'état 7 favorise des résidus I, V, R et H, et l'état 4 favorise les les résidus P, S, N, L et F. Deux cycles à deux états sont formés par b3 et b8, et b3 et b6. Les préférences des états 6 et 8 sont similaires, mais plus marquées dans l'état 6. L'état 3 montre des préférences opposées à ces deux derniers états, ce qui pourrait modéliser l'alternance polaire/apolaire des brins situés en surface. Cependant, les préférences distinguent les résidus E, T, Y et W d'une part, et K, S, L, F et C d'autre part. Ces groupes mélangent résidus polaires et apolaires.

Dans le coil, l'état 7, suivant une hélice, favorise la glycine et la proline (interrupteurs d'hélice). L'enchaînement des quatre états cachés 6, 8, 10, et 9 qui permet de sortir et de rentrer dans les brins montre des préférences pour : (A,M), (D, R, A, V), (R, A) et (D, T, M, Y). Ce motif ne semble pas correspondre aux motifs de coudes β identifiés par Hutchinson et Thornton [87].

4.4.4 Performances du modèle optimal pour la prédiction de structures secondaires

Nous rapportons dans le tableau 4.9 les performances détaillées du modèle optimal en prédiction de structures secondaires. Le modèle ne montrant pas de sur-apprentissage, seules les performances sur l'ensemble de test indépendant sont rapportées.

Tous les indices de prédiction sont meilleurs qu'avec le modèle à 21 états (voir tableau 4.6). Cependant, la prédiction des brins β reste assez mauvaise. La sensibilité est d'environ 50% : la moitié seulement des résidus en brin sont effectivement prédits comme tels.

TAB. 4.9 – Performances du modèle optimal sur les séquences de test

	Prédiction par classe				Prédiction globale	
	Sensibilité	Spécificité	CCM	Score SOV	Score Q_3	Score SOV
Hélice	74.8%	72.88%	0.60	72.6%	68.1%	64.7%
Brin	51.9%	64.55%	0.48	61.1%		
Coil	70.79%	65.09%	0.50	59.0%		

Le score Q_3 moyen est de 68.1% ; il est de 66.0% avec PSIPRED utilisé en mode uni-séquence (table 4.7). Les assignations de référence, utilisées jusqu'ici, sont produites par KAKSI, alors que PSIPRED a été optimisé avec DSSP. Pour comparer au mieux nos résultats à ceux des autres méthodes, des modèles à 36 états cachés ont été entraînés et testés sur les assignations produites par STRIDE et DSSP. Les transitions entre hélice et brin ont été rajoutées dans ces modèles.

Les résultats de cette évaluation sont présentés dans le tableau 4.10. Seuls les scores Q_3 calculés sur l'ensemble de test sont rapportés.

TAB. 4.10 – Performances des modèles entraînés et testés avec différents programmes d'assignation des structures secondaires

	HMM entraîné avec		
	KAKSI	STRIDE	DSSP
Test avec KAKSI	68.1%	67.1%	66.5%
Test avec STRIDE	67.5%	68.8%	68.3%
Test avec DSSP	66.4%	67.8%	67.9%

Comme attendu, de meilleures performances sont obtenues quand une même méthode est utilisée pour entraîner et tester les modèles. Dans ce cas, des performances similaires sont obtenues avec KAKSI et DSSP : environ 68% de bonne prédiction. Le Q_3 est légèrement plus élevé avec STRIDE : 68.8%. Lorsque les assignations de DSSP sont utilisées comme référence, PSIPRED fournit un Q_3 de 67.1% sur l'ensemble de test.

Lorsque des assignations différentes sont utilisées pour la prédiction et le test, l'écart maximum entre les scores calculés est de 2.5 points.

4.4.5 Score de confiance de la prédiction

La prédiction par l'algorithme *forward/backward* fournit les probabilités associées aux structures prédites pour chaque site de la protéine. Ces probabilités peuvent être utilisées pour évaluer la qualité de la prédiction.

La figure 4.16 montre le score Q_3, calculé sur l'ensemble de test indépendant, en fonction des valeurs des probabilités *a posteriori* des structures prédites.

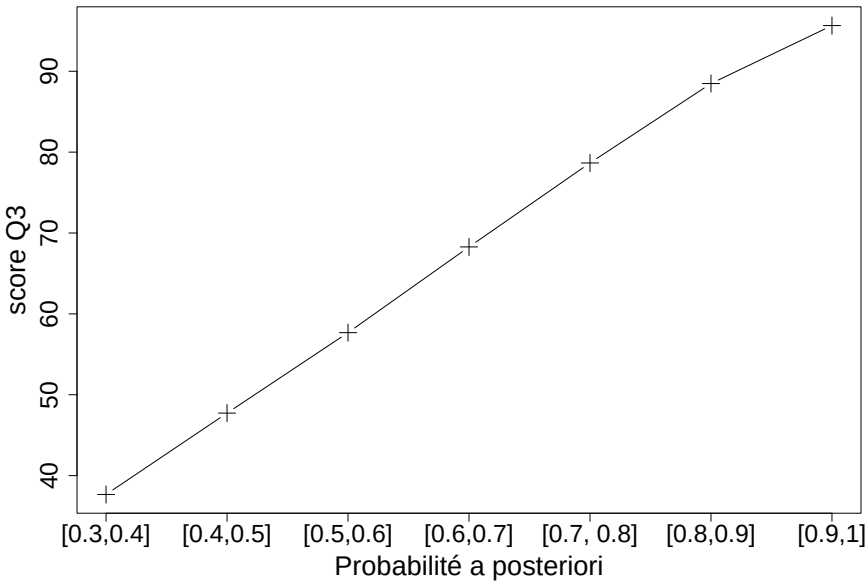

FIG. 4.16 – Score Q_3 en fonction des valeurs des probabilités *a posteriori*. Le score Q_3 est calculé sur la fraction de résidus prédits avec des probabilités dans un intervalle donné.

La qualité de prédiction et la valeurs des probabilités *a posteriori* sont très fortement corrélées. Ces probabilités peuvent être donc utilisées comme score de confiance de la prédiction.

4.4.6 Conclusion

Un modèle M1M0 construit sans *a priori* biologiques permet d'obtenir un meilleur score Q_3 que PSIPRED utilisé en mode uni-séquence sur les mêmes séquences.

En utilisant des HMM et des fenêtres dans la structure secondaire, Crooks et Brenner obtiennent un score Q_3 de 66.4% sans utiliser les profils [46], et Zheng 67.9% [209]. Les approches utilisant des HSMM atteignent des scores de 68.8% (Schmidler et al [171]) et 69.2% (Aydin et al [10]). Nous n'avons pas accès au nombre de paramètres de ces modèles qui utilisent des schémas de dépendances plus complexes que nos modèles M1M0.

Les séquences utilisées pour le test n'étant pas les mêmes que dans notre étude, les scores ne sont pas directement comparables. Néanmoins, ils sont du même ordre de grandeur. Notre modèle, avec un nombre limité de paramètres, est donc aussi performant que les autres approches utilisant les HMM.

Le point faible de notre méthode réside dans la prédiction des brins. Les performances des méthodes publiées utilisant les HMM n'étant pas aussi détaillées, nous ne savons pas si elles souffrent du même défaut que la notre. Le chapitre 5 présente une tentative visant à renforcer la prédiction des brins avec nos modèles HMM.

La prédiction de structure secondaire ne donne aucune information sur la conformation des résidus en coil. L'approche utilisée pour prédire les structures secondaires par un HMM peut très facilement être adaptée pour intégrer une information structurale sur le coil. La section suivante présente des modèles dédiés à la prédiction des angles dièdres.

4.5 Prédiction des angles dièdres à l'aide des HMM

Le travail sur les modèles HMM pour la prédiction de structure secondaire a montré que des modèles M1M0 construits sans *a priori* permettent d'obtenir de bonnes performances. De plus, il n'y a pas de sur-apprentissage avec des modèles ayant moins de 75 états cachés.

4.5.1 Zones d'angles dièdres

Pour inclure dans notre prédiction une information sur la conformation des résidus en coil, une description des structures en terme de zones d'angles dièdres est introduite. D'après la distribution des angles dièdres dans la carte de Ramachandran, 3 zones d'angles

sont distinguées. Elles sont illustrées sur la figure 4.17.

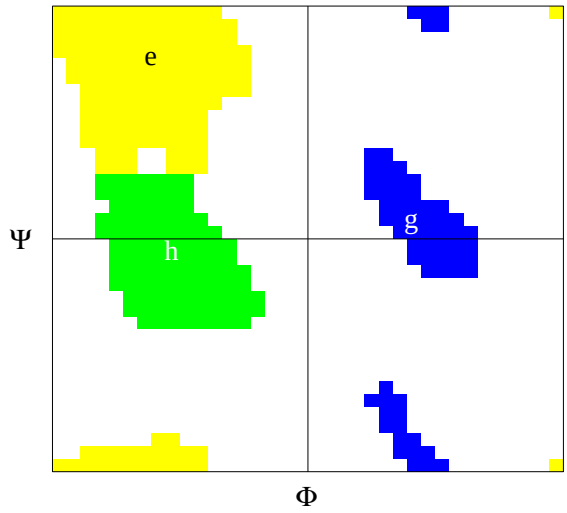

FIG. 4.17 – Définitions des 3 zones d'angles distinguées dans la carte de Ramachandran. Ces trois zones décrivent plus de 98% des résidus dans nos données. Les résidus qui ne correspondent pas à l'une de ces trois zones sont attribuées à la zone la plus proche.

Après recodage des structures en zones d'angles, 43 % des résidus correspondent à la zone e, 51% à la zone h et 6 % à la zone g. Compte tenu de cette répartition, une prédiction aléatoire obtiendrait un score Q_3 de 45%.

4.5.2 Modèle pour les zones d'angles

Le modèle utilisé est un modèle M1M0 avec 25 états cachés pour la zone h, 25 pour la zone et 15 pour la zone g. L'estimation utilise le formalisme des *class HMM*. Les paramètres sont estimés par EM sur les données recodées en zones d'angles.

Sur les 4225 transitions autorisées au départ, 1835 sont non nulles à la fin de l'apprentissage (43%). Le nombre de paramètres total est de 3005.

La structure du modèle obtenu ne sera pas détaillée ici.

4.5.3 Prédiction des zones d'angles

Les zones d'angles sont prédites à l'aide de l'algorithme forward/backward, en sommant les probabilités *a posteriori* des états modélisant la même zone d'angle.

Le tableau 4.11 rapporte les indices de prédiction du modèle à 65 états cachés. Le modèle ne montrant pas de sur-apprentissage, seules les performances sur les séquences de test sont rapportées ici.

TAB. 4.11 – Performances du modèle 65 états pour la prédiction des zones d'angles

Prédiction par classe

	Sensibilité	Spécificité	CCM
zone h	77.1%	75.3%	0.56
zone e	70.9%	71.50%	0.53
zone g	48.8%	57.75%	0.51

Prédiction globale

Score Q_3
72.7%

Le score Q_3 sur ces trois zones est de 72.7%.

Ces résultats ne sont pas directement comparables à ceux de HMMSTR et Yang et al, car ces méthodes utilisent directement ou indirectement, la notion de profil. Notons cependant que la méthode HMMSTR, qui utilise explicitement les profils, donne un score de 75 % sur un découpage en 3 classes similaires [36] (voir chapitre 1). Yang et al obtiennent un Q_3 de 78.7% en utilisant PSIPRED et les SVM [111] (voir chapitre 1).

La prédiction est moins bonne pour la zone g, de même que pour les prédictions publiées par Yang et al [111]. Si l'on isole les résidus en coil, le score Q_3 est de 61% (63.5% pour la méthode de Yang et al [111]).

4.6 Conclusion

Nous avons montré qu'un HMM M1M0 à 36 états cachés modélisant les structures secondaires, construit sans *a priori* donne de bonnes performances en prédiction de structures secondaires : le score Q_3 est d'environ 68%. Le nombre limité de paramètres du modèle permettent d'éviter les problèmes de sur-apprentissage. Le modèle obtenu, choisi sur des critères statistiques, met en lumière l'architecture interne des structures secondaires. Plusieurs états cachés sont spécifiques des transitions entre classes structurales. L'organisation des hélices et des brins montre une directionnalité dans le graphe d'états et le modèle des hélices montre une périodicité.

En utilisant un modèle HMM M1M0 à 65 états cachés, il est possible de prédire les 3 zones principales du diagramme de Ramachandran avec un score de bonne prédiction d'environ 73%.

La prédiction des brins β avec le modèle à 36 états cachés est relativement mauvaise, comme avec la plupart des méthodes. Le chapitre 5 de cette thèse présente une tentative d'amélioration de cet aspect de la prédiction.

Les méthodes de prédiction de structure locale utilisent l'information des séquences homologues, en général par le biais de profils de séquences. Dans le chapitre 6, nous présentons plusieurs méthodes pour utiliser l'information des séquences homologues avec nos modèles.

Chapitre 5

Tentative d'intégration d'une information à longue portée dans les modèles

La prédiction des structures secondaires par les HMM montre une faible qualité de prédiction pour les brins β. Les brins β s'organisant en feuillets, une composante non locale intervient dans la formation des brins β. Ces corrélations à longue portée ne peuvent pas être introduites directement dans les HMM. Nous avons donc envisagé la prise en compte indirecte de ces interactions à longue portée par le biais d'une fonction de score d'appariement potentiel en feuillets. Ce chapitre s'intitule *tentative*, la méthode n'ayant malheureusement pas permis d'obtenir les résultats escomptés.

5.1 Contexte et objectif

Les modèles HMM proposés dans le chapitre 3 permettent une prédiction correcte de la structure secondaire pour environ 68% des résidus. Cependant, l'analyse des prédictions montre que les brins β sont relativement mal prédits, en particulier trop peu de résidus sont prédits en β : la sensibilité de prédiction des brins est de 44.2% et la spécificité est de 64.8% (voir table 5.1). Cette faiblesse de prédiction des brins β est un travers courant dont souffrent la plupart des méthodes de prédiction de structure secondaire. Par exemple, PSIPRED, utilisé en mode uniséquence, obtient une sensibilité de 56% pour la prédiction des brins, pour une spécificité de 60.7%.

TAB. 5.1 – Performances du modèle à 21 états cachés et de la méthode PSIPRED sur les séquences de l'ensemble de test indépendant.

Prédiction par classe par le HMM à 21 états cachés

	Sensibilité	Spécificité	CCM	Score SOV
Hélice	75.7%	67.4%	0.56	69.7%
Brin	44.2%	64.8%	0.44	52.6%
Coil	68.3%	64.3%	0.48	56.56%

Prédiction globale

Score Q_3	Score SOV
65.7%	60.4%

Prédiction par classe par PSIPRED

	Sensibilité	Spécificité	CCM	Score SOV
Hélice	69.7%	71.3%	0.55	71.1%
Brin	56.1%	60.7%	0.48	62.9%
Coil	68.3%	64.0%	0.48	55.6%

Prédiction globale

Score Q_3	Score SOV
66.0%	62.7%

Les assignations de structure secondaire utilisées ici ne tenant pas compte des brins β isolés, l'existence d'un brin β implique l'existence d'un autre brin partenaire ailleurs dans

la structure. Cette propriété devrait pouvoir être utilisée dans la prédiction. Pour cela, il est nécessaire de former une fonction de score, qui quantifie l'existence de partenaires potentiels pour la formation de brins β.

Dans le cadre de la prédiction avec un HMM, cette information peut être utilisée de deux façons :

- une utilisation directe dans la prédiction. Cette approche nécessite de modéliser les protéines par un modèle HMM dont la structure cachée est la structure secondaire, et dont l'observation est bidimensionnelle : la séquence protéique et une séquence indicatrice de l'existence, ailleurs dans la séquence, d'un partenaire potentiel pour la formation de feuillet β.
- une utilisation *a posteriori*. Dans ce cas, la fonction de score est utilisée pour modifier les probabilités *a posteriori* fournies par le HMM.

5.2 Propositions de fonctions de score d'appariement

Le but est de calculer, le long de la séquence, une fonction indicatrice qui devrait donner un score élevé aux segments susceptibles de former un brin si des partenaires potentiels sont trouvés ailleurs dans la séquence.

5.2.1 Fonction de score basée sur les paires de résidus face à face dans les brins β appariés en feuillets β

Il s'agit de construire une matrice de score pour détecter des appariements potentiels de la séquence contre elle-même.

La fonction de score utilisée, F_1, est la suivante :

$$score = log \frac{P_\beta(paire)}{P_{alea}(paire)}$$

avec $P_\beta(paire)$ la probabilité de la paire de résidus face à face dans des feuillets β, et $P_{alea}(paire)$, la probabilité de la paire attendue aléatoirement dans une protéine. Ces probabilités sont estimées grâce aux fréquences observées.

Deux fonctions de score sont calculées selon l'orientation des brins au sein des feuillets : une fonction de score est appliquée à la recherche des appariements parallèles et une

fonction est appariée à la recherche des appariements antiparallèles.

Ces scores sont utilisés pour rechercher des appariements potentiels au sein d'une protéine : des alignements locaux sans gaps de la séquence contre elle-même sont recherchés par programmation dynamique. Les diagonales de longueur minimales et de scores positifs sont alors conservées.

Exemple de détection des appariements sur une protéine

La figure 5.1 illustre un exemple de recherche d'appariements dans la séquence de la protéine 1c22A, contenant des brins β antiparallèles. Les scores d'appariement antiparallèles F_1 sont utilisés pour détecter des diagonales lors de l'alignement de la séquence contre elle-même.

Cet exemple montre que cette fonction de score basique détecte beaucoup de faux positifs : un grand nombre de diagonales ne correspondant pas à des appariements réels ont des score positifs. Cette faible spécificité n'est pas gênante dans notre cas : notre but n'est pas de prédire les appariements, mais d'identifier des segments de séquence susceptibles de s'apparier avec d'autres.

Réduction des scores en information 1D

Pour être utilisés en tant qu'observation supplémentaire d'un HMM, l'ensemble des scores obtenus pour une séquence doivent être réduits sous forme de séquence 1D. Deux réductions sont envisagées :

- considérer, pour chaque position de la protéine, le score maximum obtenu dans la matrice,
- considérer, pour chaque position de la protéine, le score moyen obtenu dans la matrice.

Dans l'idéal, le score réduit doit être supérieur pour les résidus en β. Les réductions obtenues le long de la séquence de la protéine 1c22A sont illustrées dans la figure 5.2.

Discrimination de la fonction de score F_1

Si cette fonction de score était très discriminante, les résidus impliqués dans des brins β obtiendraient des scores supérieurs aux autres. Pour évaluer le pouvoir discriminant de la fonction de score, les scores réduits sont recueillis pour toutes les séquences d'un ensemble

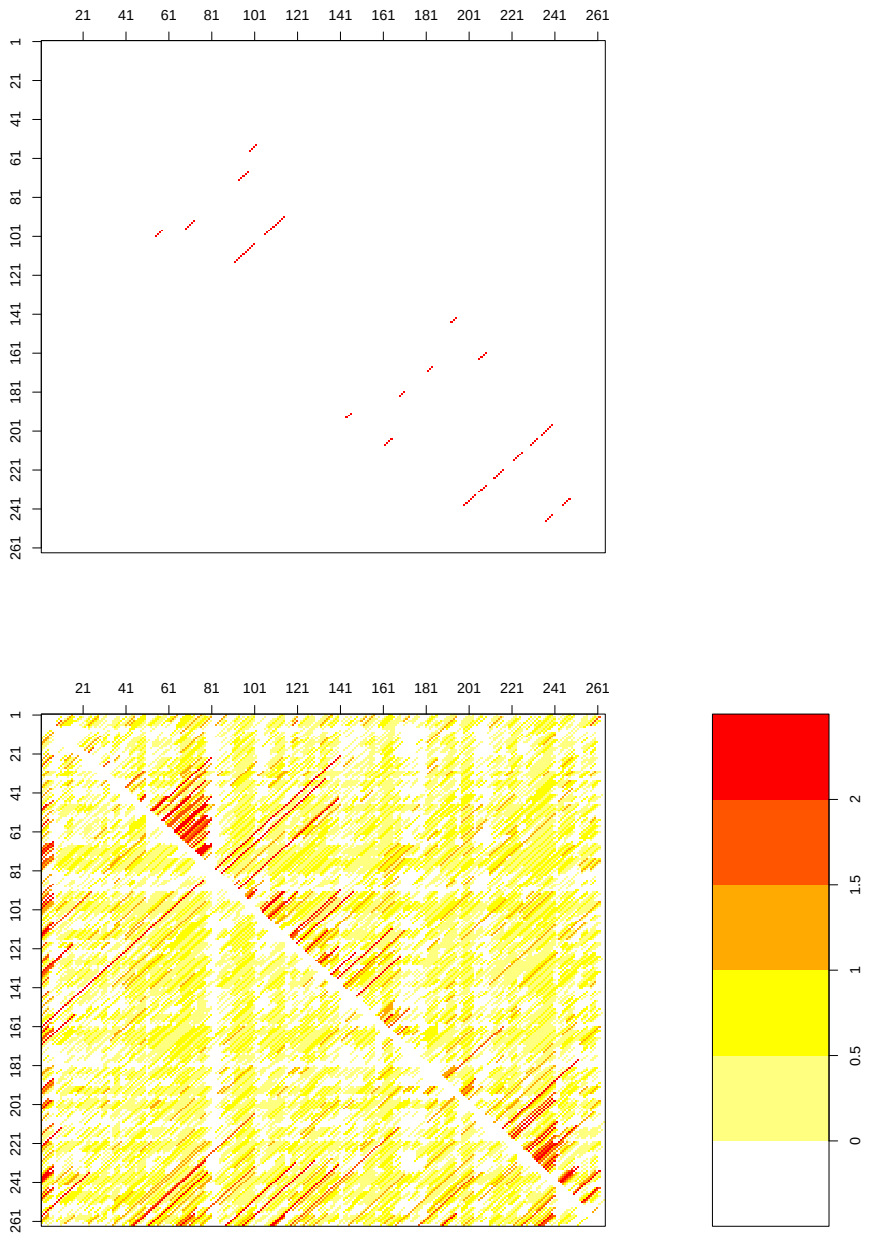

FIG. 5.1 – Exemple de recherche d'appariements dans la séquence de la protéine d2c22A. En haut, appariements observés dans la structure 3D. En bas : appariements détectés par la fonction de score antiparallèle F_1.

de test. Dans ce cadre, la nature des appariements recherchés (parallèles ou antiparallèle) n'est pas spécifiée. Les diagonales sont donc calculées pour les deux orientations possibles, en utilisant les deux matrices de score. La réduction (score maximum ou score moyen) est

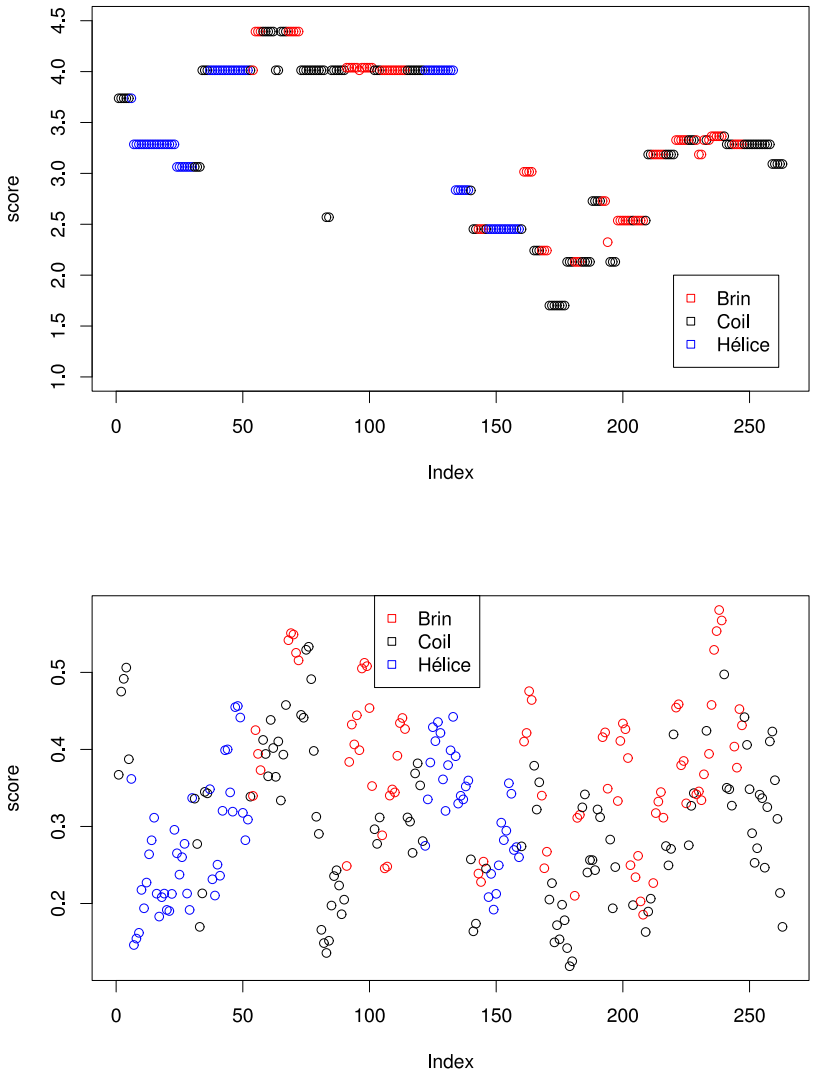

FIG. 5.2 – En haut : score maximum obtenu le long de la séquence 1c22A, obtenu avec la fonction de score F_1. En bas : score moyen obtenu le long de la séquence 1c22A, obtenu avec la fonction de score F_1. Les couleurs des points indiquent la structure secondaire du résidu.

réalisée en considérant les deux matrices.

La figure 5.3 montre les distributions des scores obtenus, distingués selon la structure secondaire (β/non-β) des résidus impliqués, calculées sur les séquences d'un ensemble de test.

La séparation des distributions de scores est meilleure en utilisant les scores moyens qu'en utilisant les scores maximums. Cependant, cette séparation semble insuffisante pour

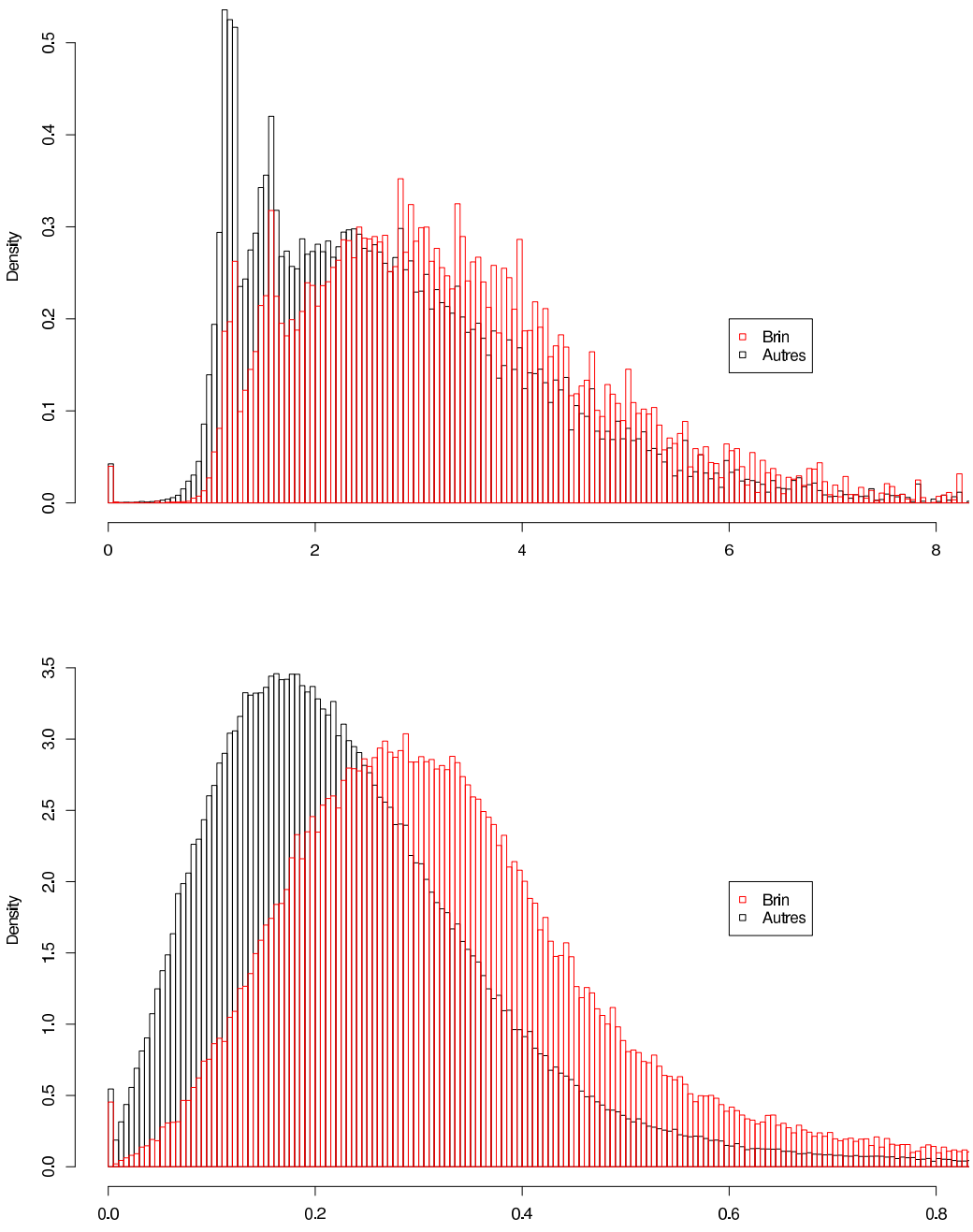

FIG. 5.3 – En haut : répartition des scores maximums obtenus sur un ensemble de séquences lors de la recherche d'appariements à l'aide de la fonction de score F_1. En bas : répartition des scores moyens obtenus sur un ensemble de séquences lors de la recherche d'appariements à l'aide de la fonction de score F_2.

que cette information puisse être intégrée dans la prédiction. Une fonction de score plus élaborée, utilisant les triplets d'acides aminés, est donc proposée.

5.2.2 Fonction de score basée sur les probabilités de triplets appariés dans les brins β

Au lieu de considérer uniquement les résidus face à face, l'identité des résidus voisins est considérée en calculant la probabilité de triplets appariés dans des feuillets β. La fonction de score proposée, F_2, est :

$$score = log \frac{P_\beta(abc, a'b'c')}{P_{alea}(abc, a'b'c')}$$

avec $P_\beta(abc, a'b'c')$ la probabilité du triplet d'acides aminés abc face au triplet $a'b'c'$ dans les feuillets β, et $P_{alea}(abc, a'b'c')$, la probabilité attendue aléatoirement.

En raison de la quantité de données disponibles, il n'est pas possible de calculer directement $P_\beta(abc, a'b'c')$. En appliquant les multiplicateurs de Lagrange, il vient :

$$P_\beta(abc, a'b'c') = P_\beta(bb') \times P_\beta(a \mid b') \times P_\beta(a' \mid b) \times P_\beta(c \mid b') \times P_\beta(c' \mid b)$$

Le calcul des probabilités associées aux triplets nécessite donc de calculer des probabilités de paires de résidus face à face ($P_\beta(bb')$) et des probabilités conditionnelles ($P_\beta(a \mid b')$, $P_\beta(a' \mid b)$, $P_\beta(c \mid b')$, $P_\beta(c' \mid b)$). Deux configurations sont distinguées pour le calcul des probabilités conditionnelles. Elles sont illustrées dans la figure 5.4.

La probabilité attendue aléatoirement, $P_{alea}(abc, a'b'c')$ est calculée en supposant un modèle M1 sur les séquences protéiques.

Cette nouvelle fonction de score est utilisée, comme précédemment, pour rechercher des diagonales d'appariements potentiels.

Exemple de détection d'appariements sur une protéine

La figure 5.5 montre les diagonales obtenues lors de la recherche d'appariements dans la séquence de la protéine d1c22A en utilisant les scores d'appariements antiparallèles de la fonction de score F_2.

Cet exemple montre que la détection est moins bruitée qu'avec la fonction de score basée sur les paires de résidus voisins.

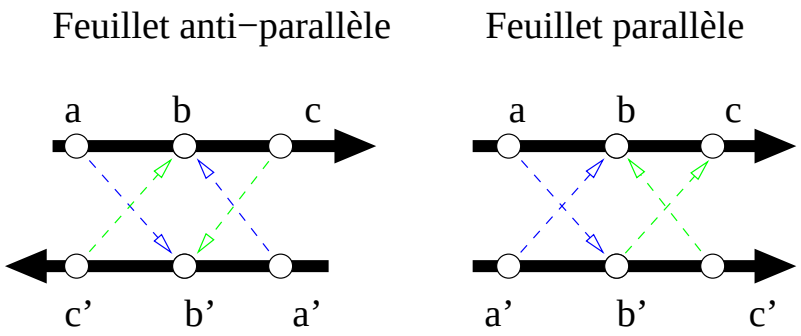

FIG. 5.4 – Appariements de triplets dans des feuillets β antiparallèles (à gauche) et parallèles (à droite). Les probabilités conditionnelles utilisées dans le calcul des probabilités de triplets sont symbolisées par des flèches en pointillés pointant vers le résidu conditionnant. Dans la configuration bleue, la probabilité du résidu x est conditionnée par le résidu face au résidu suivant x dans la séquence (probabilités $P(a \mid b')$ et $P(a' \mid b)$). Dans la configuration verte, la probabilité du résidu x est conditionnée par le résidu face au résidu précédent x dans la séquence (probabilités $P(c \mid b')$ et $P(c' \mid b)$).

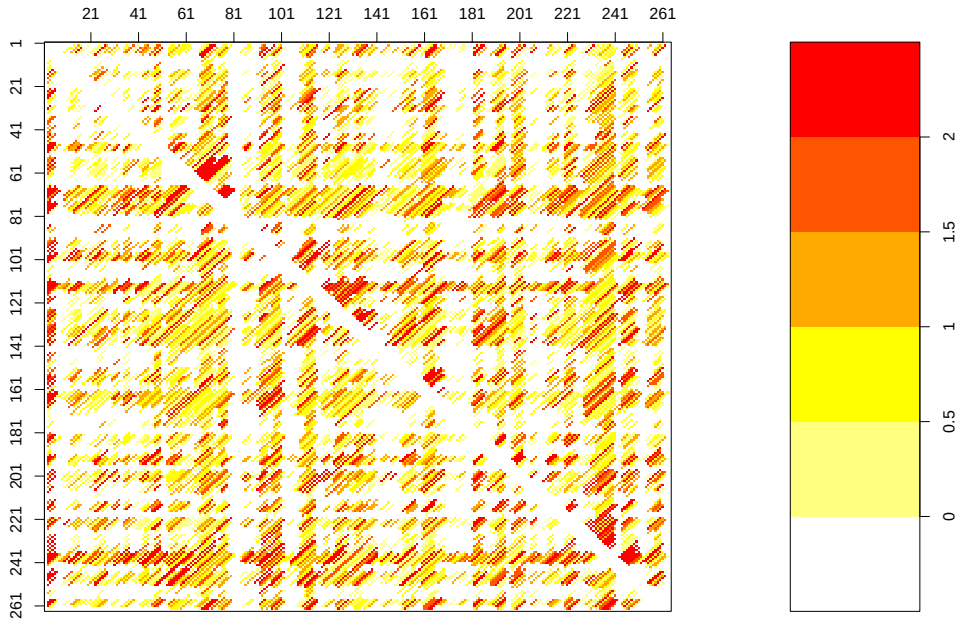

FIG. 5.5 – Appariements détectés par la fonction de score antiparallèle dans la séquence de la protéine 1c22A à l'aide de la fonction de score F_2.

Discrimination de la fonction de score

Les appariements détectés sont utilisés pour calculer des scores maximums et moyens le long des séquences, comme précédemment. Les figures 5.6 et 5.7 montrent les répartitions

obtenues pour les scores maximums et moyens.

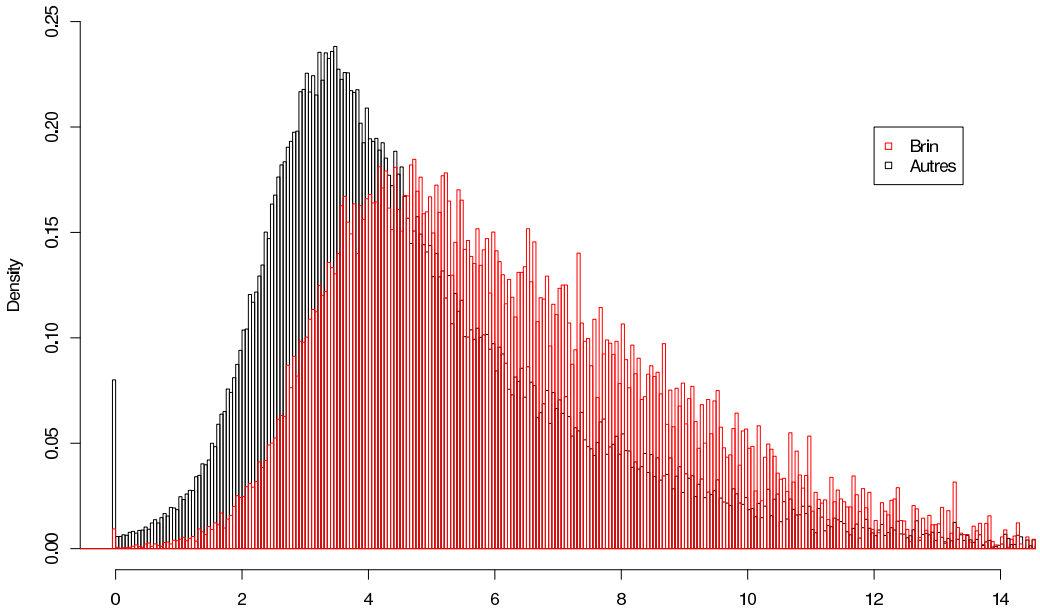

FIG. 5.6 – Répartition des scores maximums obtenus lors de la recherche d'appariements avec la fonction de score F_2, pour les résidus en β et non-β.

La séparation est meilleure en utilisant les scores moyens. Bien que la séparation soit encore assez faible, nous avons tenté d'intégrer cette information dans la prédiction par les HMM.

5.3 Utilisation directe de la fonction de score dans la prédiction par le HMM

L'information fournie par la fonction de score F_2 est utilisée en considérant un HMM qui émet indépendamment la séquence protéique et la fonction de score, comme illustré dans la figure 5.8.

Deux options sont possibles pour intégrer l'information fournie par le score :

– soit l'ensemble du modèle est estimé (i.e., les transitions entre états cachés, les lois d'émission des acides aminés et les lois d'émission des scores discrétisés),

– soit le modèle original est figé, et seules les lois d'émission des scores sont estimées.

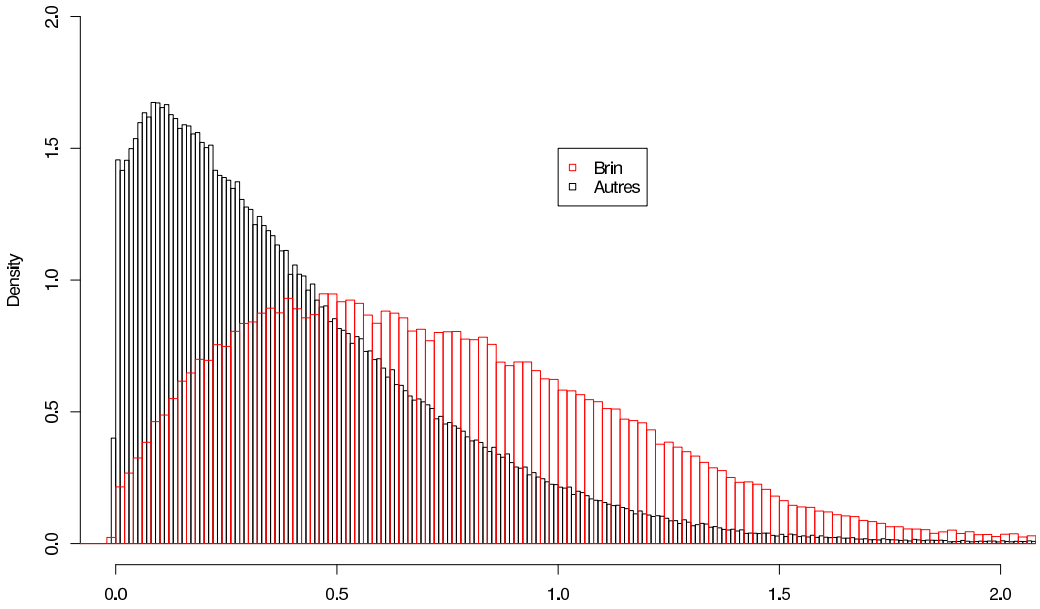

FIG. 5.7 – Répartition des scores moyens obtenus lors de la recherche d'appariements à l'aide de la fonction de score F_2, pour les résidus en β et non-β. La séparation des courbes est de 0.66 écart-type.

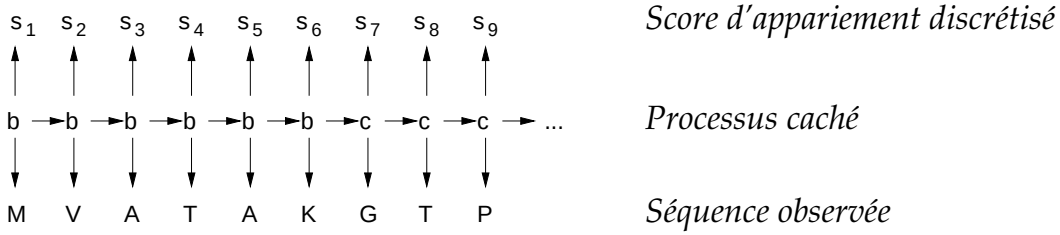

FIG. 5.8 – Intégration de la fonction de score en tant qu'observation supplémentaire du HMM. Les scores calculés sont discrétisés et traduits en séquence de caractères. Dans le cadre de HMM modélisant une classe structurale par un ensemble d'états cachés, l'observation est en réalité tridimensionnelle : la séquence de la protéine, la séquence des étiquettes de structure secondaire et les scores d'appariements sont émis indépendamment, conditionnellement au processus caché. Pour des raisons de clarté, la séquence des étiquettes n'est pas indiquée ici.

Dans les deux cas, la prédiction se fait à partir de la séquence protéique et de la séquence des scores discrétisés. Les tests effectués en utilisant le HMM à 21 états cachés,

présenté dans le chapitre 4, montrent que l'intégration de la fonction de score par ces deux méthode détériore le taux de prédiction.

Dans cette démarche, le score et la séquence protéique sont traités comme des émissions indépendantes conditionnellement au modèle caché. Or ces deux informations sont corrélées. Des normalisations des scores ont été testées pour tenir compte de cette dépendance.

Normalisation des scores

Trois normalisations sont proposées :

– normalisation 1

$$score_{norm}(A) = \frac{score(A) - \mu(A)}{sd(A)}$$

– normalisation 2

$$score_{norm}(A) = \frac{score(A) - \mu(A \mid \alpha)}{\mu(A \mid \beta) - \mu(A \mid coil)}$$

– normalisation 3

$$score_{norm}(A) = \frac{score(A) - \mu(A \mid \alpha)}{\mu(A \mid \beta) - \mu(A \mid \alpha)}$$

$\mu(A)$ et $sd(A)$ désignent respectivement la moyenne et l'écart-type des scores observés pour le résidu A, quelque soit la structure secondaire. $\mu(A \mid x)$ et $sd(A \mid x)$ désignent les mêmes quantités, pour le résidu A observé dans la structure secondaire x.

Les normalisations 2 et 3 sont issues de l'étude de la répartition des scores pour les trois classes de structures secondaires. Ces trois normalisations ne permettent cependant ni d'améliorer la discrimination des scores, ni d'améliorer la prédiction lorsque les scores sont utilisés avec le modèle à 21 états cachés.

5.4 Utilisation *a posteriori* de la fonction de score pour modifier la prédiction fournie par le HMM

Une dernière tentative a été menée pour modifier les probabilités *a posteriori* fournies par le HMM à 21 états cachés.

Le théorème de Bayes permet d'écrire :

$$P(structure/score) = \frac{P(score/structure) \times P(structure)}{P(score)}$$

ou $P(structure)$ est la probabilité *a posteriori* fournie par le modèle, et $P(score/structure)$ et $P(score)$ sont données par les distributions de scores calculées par ailleurs sur un ensemble de séquences.

Cette modification *a posteriori* permet de détecter plus de brins β que le modèle utilisé seul. La proportion des résidus impliqués dans des brins est de 22%, elle est de 16% dans les prédictions fournies par le HMM et de 23% en utilisant une correction par le score (voir tableau 5.2). Cependant, le score Q_3 global est alors de 64.2% alors qu'il supérieur à 65% avec le modèle seul. La prise en compte de la fonction de score dégrade la sensibilité de prédiction des hélices et du coil.

TAB. 5.2 – Performances du modèle à 21 états cachés seul ou en corrigeant les probabilités a posteriori à l'aide la fonction de score F_2.

HMM 21 états cachés

	Prédiction par classe				Prédiction globale
	Sensibilité	Spécificité	CCM	Proportion [a]	Score Q_3
Hélice	75.7%	67.4%	0.56	43%	65.3%
Brin	44.2%	64.8%	0.44	16%	
Coil	68.3%	64.3%	0.48	41%	

HMM 21 états cachés + correction des probabilité *a posteriori*

	Prédiction par classe				Prédiction globale	
	Sensibilité	Spécificité	CCM	Score SOV	Score Q_3	Score SOV
Hélice	70.3%	68.7%	0.56	40%	64.1%	60.4%
Brin	54.7%	54.04%	0.47	23%		
Coil	62.7%	64.6%	0.49	37%		

[a]proportion de résidus prédits dans chaque classe

5.5 Discussion

De nombreuses études ont été publiées essayant d'évaluer la spécificité d'appariement des brins β lors de la formation des feuillets. Les études expérimentales de mutations sur des résidus se faisant face dans des feuillets [207, 128] fournissent des résultats difficiles à interpréter. Par exemple, Zaremba et Gregoret [207] ont montré que des mutations portant

sur des paires de voisins enfouis réduisent la stabilité de la protéine CspA de E coli, de manière plus marquée que des mutations portant sur des paires de résidus exposés. L'effet de ces mutations peut être le fait du statut enfoui/exposé des résidus modifiés, plutôt que celui de la modification de paires spécifiques. Les études statistiques visant à évaluer la spécificité des paires de résidus dans des brins appariés [202, 88] font souvent la distinction entre les paires liées par des liaisons hydrogènes (paires HB) et paires non liées (paires NHB). Pour Wouters et Curmi [202], cette distinction paraît nécessaire pour dégager de l'information. Ils isolent ainsi 9 paires HB et 5 paires NHB significativement corrélées dans les feuillets antiparallèles. Une étude similaire de Hutchinson et Sessions [88] arrive à la conclusion que les différences entre paires HB et paires NHB sont limitées.

Lors de la construction de notre fonction de score utilisant les paires de voisins, nous avons noté peu de spécificité dans la formation des paires (résultats non rapportés ici).

Un certain nombre de travaux visent à discriminer l'appariement natif parmi des appariements alternatifs [7, 210, 13, 185]. Ainsi, Steward et Thornton 2002 [185] utilisent la théorie de l'information. La longueur de l'appariement, l'orientation des brins sont connues. L'un des brins est fixé et l'autre est décalé jusqu'à 10 positions par rapport à l'appariement réel. L'appariement correct est choisi dans 32 à 37% des cas seulement.

Enfin, la prédiction des appariements de brins β a été utilisée pour améliorer la prédiction des structures secondaires [110, 70]. Krogh et Riis [110] effectuent une prédiction en plusieurs étapes : d'une part, un réseau de neurones prédit la structure secondaire, d'autre part, des réseaux de neurones prédisent les appariements potentiels (un réseau pour les appariements parallèles et un pour les appariements antiparallèles). Les résultats sont combinés dans une fonction d'énergie. L'utilisation des réseaux de neurone prédisant les appariements n'améliore que très faiblement la prédiction des structures secondaires : en sortie du premier réseau, le score Q_3 est de 66.3% ; il est de 66.5% après combinaison des résultats. La méthode de prédiction PREDATOR, de Frishman et Argos [70] tient compte des appariements potentiels en calculant des scores selon une méthode proche de celle que nous avons utilisée. Différentes statistiques sont utilisées pour les paires HB et NHB. Les diagonales potentielles sont calculées en utilisant le motif de liaison hydrogène donnant le score maximum. Elles sont réduites en information 1D en retenant le score maximum. Ces scores de préférences sont utilisés en combinaison avec une méthode des plus proches voisins et des règles de décision. L'effet de l'utilisation des scores d'appariements sur le score

Q_3 n'a pas été quantifié. Des études plus récentes [172, 42], utilisant les HSMM, intègrent les corrélations longue distance pour détecter les appariements potentiels. Schmidler et al n'ont pas quantifié l'apport de ces corrélations pour la prédiction de structure secondaire. Pour Chu et al, cette prise en compte n'améliore pas significativement la prédiction.

5.6 Conclusion

L'intégration de corrélations à longue portée dans la prédiction des brins β est donc une tâche difficile.

La formation des appariements dans les brins β semble assez peu spécifique, aussi les méthodes de prédiction des motifs d'appariements donnent des résultats relativement modestes. La détection d'appariements potentiels est une tâche moins compliquée, puisque le niveau de prédiction est moins détaillé. Certaines méthodes de prédiction de structure secondaire utilisent la détection de ces appariements, mais l'apport de cette information supplémentaire semble assez mineur. Nous avons essayé d'utiliser une information extérieure pour améliorer les prédictions fournies par nos HMM. Il semble que l'obtention de cette information constitue en elle-même une tâche difficile et que la prédiction par les HMM n'est pas robuste par rapport à des informations supplémentaires de faible qualité.

Chapitre 6

Utilisation des séquences homologues dans la prédiction

Dans ce chapitre, nous présentons une amélioration de notre prédiction de la structure secondaire des protéines ou des zones d'angles par la prise en compte des séquences homologues.

La combinaison des prédictions réalisées indépendamment sur un ensemble de séquences alignées permet d'obtenir un score Q_3 de 75.9% pour la prédiction des structures secondaires et de 78% pour les zones d'angles.

Une méthode de couplage du HMM avec un arbre phylogénétique est proposée. Malgré son cadre méthodologique séduisant, cette méthode ne permet pas d'améliorer la prédiction. Une étude menée sur des séquences simulées permet néanmoins de valider cette approche dans une situation idéale. Ce couplage suppose que la séquence d'états cachés sous-jacente est commune à toutes les séquences d'une famille. L'étude sur séquences simulées montre que cette hypothèse est fondamentale. Le défaut de cette approche sur séquences réelles est probablement due à la violation de cette hypothèse.

6.1 Contexte et objectif

Les protéines issues d'un ancêtre commun (protéines homologues) évoluent par mutations aléatoires des séquences. La pression de sélection s'exerce sur le maintien de la fonction des protéines. La fonction étant intimement liée à la structure 3D, la séquence n'est *autorisée* muter que dans la mesure où la structure est conservée. La conséquence de cette évolution sous contrainte est que des séquences différentes peuvent correspondre à la même structure. Même lorsqu'aucune séquence de protéine homologue n'est disponible pour dériver un modèle 3D, il n'est pas rare que les banques de séquences contiennent de nombreuses séquences homologues de la séquence à prédire. Ces séquences homologues reflètent la pression évolutive sur une famille de séquences. Les sites catalytiques des enzymes sont par exemple très conservés alors que les boucles situées à la surface de la structure le sont moins. D'une manière plus globale, les mutations doivent conserver la structure fonctionnelle des protéines.

Pour revenir à un aspect plus pratique, lors du développement d'une méthode d'apprentissage de la structure locale, les séquences homologues constituent un ensemble plus étendu d'exemples de séquences ayant la même structure. Ainsi, l'utilisation des séquences homologues a permis d'améliorer considérablement les performances des méthodes de pré-

diction de structure secondaire, voir par exemple [161, 154, 90].

L'information contenue dans les familles de séquences est très souvent prise en compte sous la forme de profils, constitués des fréquences d'acides aminés calculées sur les colonnes d'un alignement multiple. Les systèmes basés sur les réseaux de neurones ou les SVM peuvent être entraînés sur des fenêtres glissantes le long des profils [161, 154, 90, 103, 84, 198, 83]. Dans le cas de méthodes basées sur des HMM, les émissions de profils sont modélisées par une distribution multinomiale [191, 36, 38] ou par un modèle de grandes déviations [46].

Une autre approche consiste à combiner les prédictions indépendantes obtenues sur les séquences d'une famille [154, 93]. Il est nécessaire d'utiliser un système de pondération pour tenir compte des corrélations entre les séquences. En effet, les banques de séquences n'offrent pas un échantillon aléatoire équilibré des séquences. Une recherche d'homologues peut fournir une collection de séquences très proches ainsi que quelques séquences plus distantes. La combinaison doit tenir compte de ce déséquilibre afin de ne pas surestimer l'information apportée par les séquences les plus représentées.

Vingron et al [196] résument ainsi les propriétés souhaitées pour un système de pondération :

- les séquences identiques doivent avoir un poids égal,
- si l'alignement contient n séquences identiques de poids w, le retrait de $n-1$ de ces séquences doit résulter en la définition d'un poids nw sur la séquence restante,
- des séquences fortement similaires doivent avoir des poids plus faibles que des séquences divergentes,
- la méthode de pondération ne doit pas faire usage d'information inutile, et ne doit pas supprimer d'information nécessaire.

Différents schémas de pondération ont été proposés. Une pondération simple, due à Henikoff et Henikoff [81] consiste à *partager* la contribution des séquences portant le même acide aminé en un site considéré. Une pondération basée sur les arbres phylogénétiques [56] repose sur une interprétation intuitive de l'arbre comme parcouru par un courant électrique ; les intensités mesurées en sortie des feuilles sont égales aux poids recherchés. D'autres schémas de pondérations sont plus complexes à mettre en œuvre. Par exemple, la pondération de Sibbald et Argos [177], basée sur les cellules de Voronoï, nécessite un échantillonnage de l'espace des séquences par des méthodes de Monte-Carlo. Pour un aperçu

des différentes méthodes de pondération, on pourra consulter par exemple [56, 196].

Notre but est de pouvoir réutiliser les modèles HMM établis pour la prédiction sur séquences seules. Différents schémas de pondérations ont été utilisés pour combiner les prédictions indépendantes sur les séquences d'une famille. Une méthode de couplage du HMM avec l'arbre phylogénétique est également proposée.

6.2 Matériel et méthodes

6.2.1 Données

Les données utilisées sont les mêmes celles du chapitre 4 : 2530 séquences non redondantes, réparties en 2024 séquences pour la validation croisée et 506 séquences de test. Pour chacune des séquences, les séquences homologues sont rapatriées par une recherche avec PSI-BLAST dans une banque de séquences dont la redondance est réduite à 80% d'identité entre séquences.

Un profil PSI-BLAST est construit itérativement. A la première itération de la recherche, les homologues de la séquence cible sont détectés en utilisant la matrice de score BLOSUM62. Les séquences détectées sont utilisées pour construire un profil qui permet de dériver des scores spécifiques pour chaque position de la séquence cible (*PSSM pour Position Specific Scoring Matrix*) Ce profil est modifié à chaque itération d'après les résultats de la comparaison. La puissance de PSI-BLAST, outre l'efficacité de la procédure itérative, réside dans le calcul de *e-value* (*expectation value*, littéralement valeur attendue) qui évalue la significativité des scores associés à chaque séquence détectée.

Le seuil de e-value est fixé à 10^{-3} et le nombre maximum d'itérations à 5. Cette recherche génère en moyenne 196 séquences par famille.

Les séquences sont filtrées pour ne conserver que les séquences ayant au moins 30% d'identité de séquence avec la séquence requête, et au plus 20% de gap. Après ce traitement, le nombre moyen de séquences par famille est de 59.

Les arbres phylogénétiques sont calculés par l'algorithme Neighbour Joining (NJ) sur les distances de Kimura, en utilisant le package GCG. GCG produit un arbre sans racine dont les longueurs de branche sont exprimées en nombre de mutations pour 100 résidus.

6.2.2 Systèmes de pondération des séquences

La façon la plus simple de prendre en compte l'information d'une famille de séquences est de réaliser des prédictions indépendantes sur chaque séquence. La structure secondaire consensus est le résultat de la combinaison de ces prédictions individuelles. Pour tenir compte des corrélations entre séquences, il est nécessaire d'utiliser des pondérations. Au final, la probabilité associée à la structure *struct*, pour une famille de N séquence est donnée par :

$$P(S_t = struct/famille) = \sum_{i=1}^{N} w_i P(S_t = struct/X^i)$$

où $P(S_t = struct/X^i)$ est la probabilité *a posteriori* de la structure *struct* pour la séquence i, calculée par l'algorithme forward/backward, et w_i est le poids associé à la séquence i. L'interprétation de cette formule est assez intuitive : la prédiction finale est le résultat d'un vote, dans lequel les séquences ont des poids qui dépendent de leur représentativité dans la famille. Le vote des séquences portant des gaps n'est pas comptabilisé.

Les schémas de pondérations envisagés sont : la pondération de Henikoff [81], une pondération basée sur la topologie des arbres due à Thompson et al [190] et une pondération basée sur le partage d'information dans un arbre phylogénétique.

6.2.3 Pondération de Henikoff et Henikoff, 1994 [81]

Le calcul des pondérations de Henikoff à partir d'une famille de séquences alignées est le suivant. Le poids associé à la séquence i, pour la colonne t de l'alignement est donné par :

$$w_i^t = \frac{1}{n_{diff}^t \times n_{x_i}^t}$$

n_{diff}^t est le nombre de lettres différentes observées dans la colonne t, et $n_{x_i}^t$, le comptage de la lettre portée par la séquence i, dans la colonne t. Si, dans une colonne d'un alignement de 10 séquences, on observe 6 fois la lettre a et 4 fois la lettre b, les séquences portant la lettre a reçoivent des poids $\frac{1}{2\times 6}$ et les séquences portant la lettre b, des poids $\frac{1}{2\times 4}$. La pondération de Henikoff partage donc la contribution des séquences portant la même lettre. Les gaps ne sont pas pris en compte dans ce calcul.

Le poids de la séquence i est la moyenne des poids calculés pour les différentes colonnes de l'alignement. Ce schéma de pondération est utilisé dans le calcul des profils dans

PSI-BLAST pour corriger les fréquences observées dans les colonnes d'un alignement de séquences.

6.2.4 Pondération basée sur les arbres phylogénétiques, Thompson et al, 1994 [190]

L'arbre produit par l'algorithme NJ est enraciné en utilisant l'algorithme des poids moyens : la racine est placée en un point tel que la longueur moyenne des branches soit la même dans chaque sous-arbre. Considérons l'arbre phylogénétique illustré dans la figure 6.1. l_j désigne la longueur de la branche au-dessus du noeud j et O_j le nombre de séquences dans le sous-arbre porté par j.

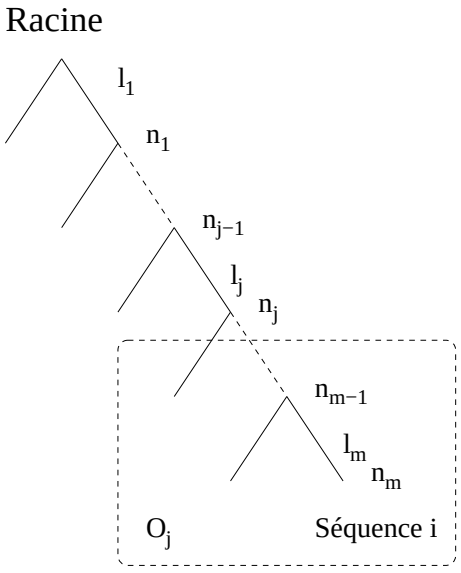

FIG. 6.1 – Représentation schématique d'un arbre phylogénétique

Le poids de la séquence i est calculé en parcourant l'arbre depuis la racine, jusqu'à la séquence i :

$$w_i = \sum_{j=1}^{m} \frac{l_j}{O_j}$$

Ces poids sont normalisés pour sommer à 1. Le poids des séquences diminue si le voisinage dans l'arbre est dense. Ce système de pondération est utilisée dans le programme d'alignement multiple CLUSTALW pour pondérer les scores de substitution [190].

6.2.5 Modèle d'évolution

Afin de mettre en place une pondération des séquences tenant compte des séquences apparaissant aux feuilles de l'arbre, il est nécessaire d'introduire un modèle d'évolution. Pour l'utilisation ultérieure en couplage avec les HMM, il est souhaitable que ce modèle soit cohérent avec les HMM utilisés qui sont de type M1M0. Le modèle d'évolution introduit est donc un modèle à sites indépendants. Dans un processus d'évolution par saut à sites indépendants, la probabilité d'observer l'acide aminé j après un temps Δt, sur un site portant initialement l'acide aminé i est donnée par :

$$\begin{cases} p_{ij}(\Delta t) \approx \alpha_{ij}\Delta t & \text{si } i \neq j \\ p_{ii} \approx 1 - \sum_{j \neq i} \alpha_{ij} t & \text{si } i = j \end{cases}$$

Les α_{ij} sont les termes d'un matrice G appelée générateur infinitésimal. Par définition $\alpha_{ii} = -\sum_{i \neq j} \alpha_{ij}$. On démontre que

$$p_{ij}(t) = (exp(Gt))_{ij}$$

$exp(Gt)$ représentant l'exponentielle de la matrice Gt.

Pour assurer la cohérence avec le HMM lors du couplage entre le modèle d'évolution et le HMM, la loi stationnaire du processus d'évolution est prise égale à la loi d'émission de l'état caché considéré. Une solution possible et simple est d'utiliser un générateur infinitésimal G_u, spécifique de l'état u, de termes :

$$\begin{cases} \alpha^u_{ij} = b_u(j) & \text{si } i \neq j \\ \alpha^u_{ij} = b_u(i) - 1 & \text{sinon.} \end{cases}$$

b_u étant la loi d'émission de l'état caché u. Dans ce cas :

$$p^u_{ij}(t) = (1 - e^{-t})b_u(j) + e^{-t}1_{\{i=j\}}$$

$1_{\{i=j\}}$ étant une indicatrice qui vaut 1 si la condition entre accolades est remplie. Cette formulation de p^u_{ij} montre que tout se passe comme sous un modèle de Poisson renouvelant :

– Les événements de mutation constituent un processus de Poisson sur l'arbre, $1 - e^{-t}$ représente la probabilité de mutation après un temps t,

$$p^u_{ij}(t) = P(\text{mutation})b_u(j) + P(\text{pas de mutation})1_{\{i=j\}}$$

En pratique, les probabilités de mutation sont directement fournies par l'arbre phylogénétique produit par GCG : les branches sont mesurées en nombre de mutations attendues pour 100 résidus.

– L'état d'un site après mutation est un renouvellement, c'est à dire que le remplacement se fait sans mémoire de l'état précédent. Sous un tel modèle, une mutation peut être invisible, si la valeur du site après mutation est la même qu'avant.

Les sites étant indépendants, le modèle peut être hétérogène spatialement : différentes lois de renouvellement (= différents états cachés) peuvent être appliquées sur des sites voisins. Ce modèle d'évolution étant réversible, les arbres phylogénétiques peuvent être enracinés arbitrairement sans que cela ne change les calculs présentés dans la suite.

6.2.6 Pondération équitable utilisant l'arbre phylogénétique

Munis d'une famille de séquences alignées et de l'arbre phylogénétique correspondant, nous proposons un nouveau système de pondération. Le principe est d'estimer les comptages *réels* des acides aminés en un site de l'alignement, compte tenu de la phylogénie des séquences, en d'autres termes, le nombre de fois où un acide aminé est apparu indépendamment dans l'arbre. Les poids des séquences sont alors déduits en partageant équitablement ces comptages entre les séquences d'après leurs situations dans l'arbre.

Un exemple simple de ce principe est illustré dans la figure 6.2. Les arbres 1 et 2 comportent tous les deux 7 séquences dont deux portent la lettre a et 5 la lettre b. Considérons le comptage *réel* de la lettre a. Dans l'arbre 1, les séquences 1 et 2 qui portent cette lettre ne sont séparées que d'un intermédiaire. Il est fort probable que ces deux séquences proches apportent des informations redondantes. Le comptage réel de la lettre a est donc probablement 1. A l'inverse, dans l'arbre 2, les séquences 1 et 6 portant la lettre a sont séparées par plusieurs intermédiaires. De plus, le fait que les autres séquences portent la lettre b indique que les séquences 1 et 6 sont probablement le fruit de mutations indépendantes. Dans ce cas, le comptage réel de a dans l'arbre est probablement 2.

Les arbres 3 et 4 permettent d'illustrer la notion de pondération équitable. Supposons connu le comptage réel de la lettre a pour ces deux arbres. Le principe de la pondération équitable est de partager ce comptage entre les séquences qui y contribuent, d'après leur situation dans l'arbre et l'observation des autres séquences. Dans l'arbre 3, les séquences qui portent cette lettre sont situées dans la même région de l'arbre. Elles se verront donc

logiquement attribuer des poids similaires Dans l'arbre 4, la séquence 6 est retrouvée dans une région où de nombreux b sont observés. Cette séquence apporte plus d'information que les séquences 1 et 2 car elle est plus *originale*. Elle recevra donc un poids plus important que 1 et 2.

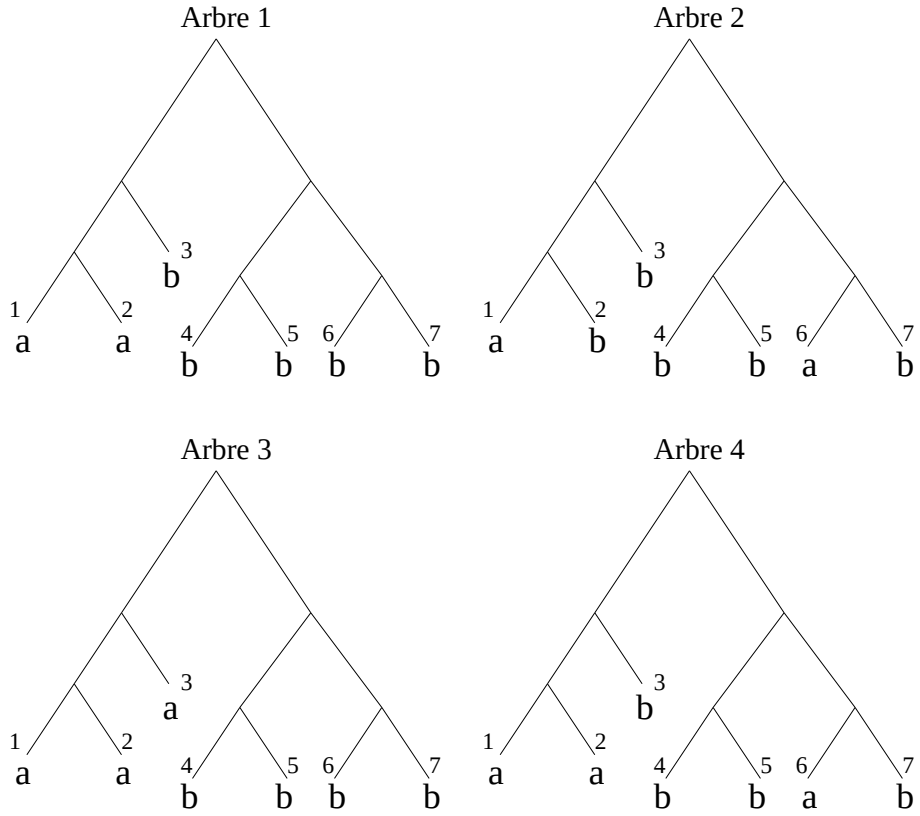

FIG. 6.2 – Représentation schématique illustrant les notions de comptages *réels* portés par un arbre et de pondération équitable (voir texte). La représentation adoptée est celle des arbres binaires enracinés, ce qui peut paraître étrange car l'algorithme NJ fournit des arbres sans racine. Le modèle d'évolution que nous utilisons est réversible et le placement de la racine est donc indifférent. Dans les calculs de comptages et de pondérations équitables, la racine ne joue pas de rôle particulier, elle est considérée comme un noeud intermédiaire quelconque.

Les sites étant indépendants, le calcul des pondérations et des espérances des comptages réels sont présentés pour une seule colonne de l'alignement. Les poids associés aux séquences sont calculés en réalisant la moyenne sur les colonnes. Il est aussi possible de conserver des pondérations locales spécifiques par position.

Après l'introduction des notations nécessaires, nous commençons par définir formellement la notion de *comptage réel* sous l'appellation d'*information conditionnelle*. Les calculs des pondérations équitables et de l'information conditionnelle sont ensuite présen-

tés.

Notations

La racine de l'arbre sera notée R.

Les noeuds de l'arbre sont notés g.

Un noeud portant une séquence est une feuille.

g_0 et g_1 désignent les noeuds fils de g.

Le sous-arbre porté par le noeud g est noté A_g (voir figure 6.3). A désigne l'arbre entier.

Deux séquences sont dites identiques par copie s'il n'y pas eu de mutation dans l'arbre sur le chemin les reliant.

$g_0 \equiv g_1$ signifie que g_0 et g_1 sont identiques par copie.

$g_0 \equiv A_{g_0}$ désigne l'événement : le sous-arbre porté par g_0 comporte une séquence identique par copie à g_0.

$A_{g_0} \equiv A_{g_1}$ désigne l'événement : le sous-arbre A_{g_0} contient une séquence identique par copie à une séquence du sous-arbre A_{g_1}.

Φ représente l'état de l'ensemble des feuilles de l'arbre, et Φ_g l'état des feuilles du sous-arbre A_g.

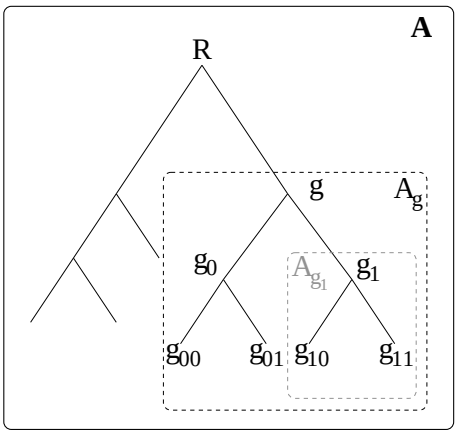

FIG. 6.3 – Représentation schématique d'un arbre phylogénétique

Définition de l'information conditionnelle portée par un arbre phylogénétique

Deux séquences qui portent la même lettre ne sont pas nécessairement identiques par copie : elles peuvent être le résultat d'une mutation invisible, ou de deux mutations indépendantes. Si elles sont identiques par copie, elles n'apportent pas plus d'information qu'une seule séquence. En revanche, si elles ne sont pas identiques par copie, elles comptent pour deux.

La relation d'identité par copie est une relation d'équivalence qui définit une partition de l'ensemble des feuilles d'un arbre phylogénétique : chaque classe d'équivalence regroupe les copies d'une même variable aléatoire. Le nombre de classes d'équivalence dans un arbre n'est pas connu car les événements évolutifs le long des branches de l'arbre ne sont pas connus.

En revanche il est possible de calculer l'espérance de ces comptages connaissant l'état des feuilles, que l'on nomme *information conditionnelle*. Cette information désigne donc le comptage corrigé d'un acide aminé dans l'arbre : le nombre d'occurrences de la lettre, corrigé par la prise en compte des corrélations entre séquences.

Calcul des pondérations équitables d'après l'information conditionnelle

Supposons que l'on connaisse l'information sur l'acide aminé a, $I_a(A_g \mid \Phi)$, portée par chaque noeud g de l'arbre. Ces informations permettent de calculer les poids associés aux séquences par un partage équitable de l'information selon la position des séquences dans l'arbre.

Considérons la séquence i, qui porte la lettre a_i. On définit, pour les noeuds de l'arbre situés entre la séquence i et la racine, les coefficients K_i^g, qui sont des pondérations réactualisées à chaque étape :

si g est la feuille de l'arbre qui porte la séquence i : $K_i^g = 1$.

si g est un noeud interne situé entre la feuille portant la séquence i et la racine :

$$K_i^g = K_i^{g_{prec}} \frac{I_{a_i}(A_g \mid \Phi)}{I_{a_i}(A_{g0} \mid \Phi) + I_{a_i}(A_{g1} \mid \Phi)}$$

$K_i^{g_{prec}}$ est le coefficient calculé pour le noeud fils de g situé sur le trajet entre la séquence i et la racine.

Le poids associé à la séquence i est donné par le coefficient K de la racine :

$$w_i = K_i^R$$

On a $\sum w_i = I_a(A \mid \Phi)$.

Ces coefficients permettent de partager équitablement l'information $I_a(A_g \mid \Phi)$ entre toutes les séquences qui portent la même lettre a, compte tenu de leur situation dans l'arbre. Ces pondérations seront donc appelées *pondérations équitables*.

Lors du calcul du poids d'une séquence, l'arbre est parcouru depuis cette séquence jusqu'à la racine. Supposons que lors du calcul, on atteigne un noeud dont les fils sont des sous-arbres fortement corrélés, c'est à dire qu'il y a de fortes chances pour que ces deux sous-arbres portent des séquences identiques par copie. Dans ce cas, l'information portée par ce noeud est nettement inférieure à la somme des informations portées par les deux sous-arbres : en effet, si ces deux sous-arbres sont très corrélés, ils apportent des informations redondantes. Le nouveau coefficient calculé va donc être d'autant plus faible, puisqu'il est proportionnel à $\frac{I_{a_i}(A_g|\Phi)}{I_{a_i}(A_{g0}|\Phi)+I_{a_i}(A_{g1}|\Phi)}$. Une séquence provenant d'une partie *dense* de l'arbre recevra donc un poids moins important.

Le calcul des informations fait intervenir la loi de renouvellement des acides aminés. Deux utilisations de ce schéma de pondération sont possibles :

1. considérer une seule loi de renouvellement correspondant à la fréquence des acides aminés dans la base de données. Ces pondérations seront utilisées classiquement pour combiner les prédictions.

2. considérer des lois de renouvellement spécifiques de chaque type de structure secondaire. Dans le cadre de la prédiction, la structure secondaire n'étant pas connue, trois jeux de pondérations seront calculés, et utilisés pour combiner les prédictions pour chaque classe.

Dans ce calcul, les gaps ont été pris en compte, en supposant une fréquence de gap de 5% dans la loi de renouvellement. Les poids sont normalisés à 1, et le poids d'une séquence est la moyenne sur les poids calculés à chaque position.

Calcul de l'information conditionnelle $I_a(A_g \mid \Phi)$

Si les séquences étaient toute indépendantes, cette information conditionnelle serait égale au nombre de séquences portant la lettre a. Les séquences étant corrélées, le *vrai* comptage est en réalité inférieur au nombre de séquences portant cette lettre.

L'information portée par l'arbre, relative à la lettre a, est donnée par :

$$I_a(A \mid \Phi) = n(a) - \sum_{g \in G} P[g_o \equiv A_{g_0}, g_1 \equiv Ag_1, g_0 \equiv g_1, g = a \mid \Phi]$$

$n(a)$ désigne le nombre de séquences portant la lettre a dans l'arbre. G désigne l'ensemble des noeuds internes de l'arbre, plus la racine.

Justification Considérons un arbre A, constitué de deux sous-arbres A_0 et A_1. L'information portée par cet arbre est la somme des informations portées par ses sous-arbres fils, dont il faut soustraire la probabilité qu'au moins l'une des séquences de A_0 soit identique par copie à l'une des séquences de A_1 :

$$I_a(A \mid \Phi) = I_a(A_0 \mid \Phi) + I_a(A_1 \mid \Phi) - P[g \equiv A_0, g \equiv A_1, g_0 \equiv g_1, g = a \mid \Phi]$$

Si une séquence de A_0 est identique par copie à une séquence de A_1, cela implique que les noeuds intermédiaires g_0 et g_1 sont identiques par copie à la racine de l'arbre. De même, pour les sous-arbres fils A_0 et A_1 :

$$I_a(A_0 \mid \Phi) = I_a(A_{00} \mid \Phi) + I_a(A_{01} \mid \Phi) - P[g_0 \equiv A_{00}, g_0 \equiv A_{01}, g_{00} \equiv g_{01}, g_0 = a \mid \Phi]$$

$$I_a(A_1 \mid \Phi) = I_a(A_{10} \mid \Phi) + I_a(A_{11} \mid \Phi) - P[g_1 \equiv A_{10}, g_1 \equiv A_{11}, g_{10} \equiv g_{11}, g_1 = a \mid \Phi]$$

En développant cette formule pour tous les noeuds fils, on arrive à la formule encadrée ci-dessus.

Si l'on note :

$$X_a^g = P[g_o \equiv A_{g_0}, g_1 \equiv Ag_1, g_0 \equiv g_1, g = a \mid \Phi]$$

la probabilité que le noeud g porte la lettre a et que les arbres A_0 et A_1 issus de g aient au moins une feuille identique par copie à g, conditionnellement à l'état des feuilles de l'arbre ;

$$Q_a^{g_0} = P(g_0 \equiv A_{g_0} \mid g = a, \Phi_{g_0})$$

la probabilité que le noeud g_0 soit représenté dans le sous-arbre A_{g_0}, sachant $g = a$, et l'état des feuilles ;

$$P_a^{g_0} = P(g_0 \equiv g \mid g = a, \Phi_{g_0})$$

la probabilité que g_0 soit identique par copie à son père, sachant $g = a$ et l'état des feuilles ;

$$\Pi_a^g = P(g = a \mid \Phi)$$

la probabilité d'avoir $g = a$ sachant l'état de l'ensemble des feuilles ; on a alors :

$$I_a = n(a) - \sum_{g \in G} X_a^g$$

$$X_a^g = Q_a^{g_0} \times P_a^{g_0} \times Q_a^{g_1} \times P_a^{g_1} \times \Pi_a^g$$

Calcul des termes $Q_a^{g0} = P(g_0 \equiv A_{g_0} \mid g = a, \Phi_{g_0})$

$$\begin{cases} Q_a^g = 1_{\{g=a\}} & \text{si g est une feuille} \\ Q_a^g = 1 - (1 - P_a^{g_0} Q_a^{g_0})(1 - P_a^{g_1} Q_a^{g_1}) & \text{si g est un noeud interne.} \end{cases}$$

La quantité $P_a^{g_0} Q_a^{g_0}$ est la probabilité que le sous-arbre A_{g_0} comporte une séquence identique par copie à g. La quantité $1 - P_a^{g_0} Q_a^{g_0}$ représente donc la probabilité que A_{g_0} ne contienne *pas* de séquence identique par copie à g. Le produit de ces deux termes donne ainsi la probabilité pour qu'aucun des deux sous-arbres ne contienne de séquence identique par copie à g.

Calcul des termes $P_a^{g_0} = P(g_0 \equiv g \mid g = a, \Phi_{g_0})$

$$\begin{aligned} P_a^{g_0} &= \frac{P(g_0 \equiv g \mid g = a) P(\Phi_{g_0} \mid g = a, g_0 \equiv g)}{P(\Phi_{g_0} \mid g = a)} \quad \text{théorème de Bayes} \\ &= \frac{P(g_0 \equiv g \mid g = a) P(\Phi_{g_0} \mid g_0 = a)}{P(\Phi_{g_0} \mid g = a)} \\ &= \frac{P(g_0 \equiv g \mid g = a) P(\Phi_{g_0} \mid g_0 = a)}{\sum_b P(\Phi_{g_0} \mid g_0 = b) P(g_0 = b \mid g = a)} \\ &= \frac{P(g_0 \equiv g) F_a^{g_0}}{\sum_b F_b^{g_0} P(g_0 = b \mid g = a)} \quad \text{si l'on note } F_a^g = P(\Phi_g \mid g = a) \end{aligned}$$

Le terme $P(g_0 \equiv g)$ se déduit de la longueur de la branche l reliant g à g_0 : il est égal à $1 - l$. Le terme $P(g_0 = b \mid g = a)$ est calculé grâce à la longueur de la branche, l, et la loi

de renouvellement π :

$$\begin{aligned} P(g_0 = b \mid g = a) &= 1 - l + l\pi(b) \text{ si a=b} \\ P(g_0 = b \mid g = a) &= l\pi(b) \text{ sinon} \end{aligned}$$

Si $a = b$, il peut s'agir d'une identité par copie (probabilité $1-l$), ou d'une mutation nulle (mutation avec la probabilité l, puis tirage de la lettre b d'après la loi de renouvellement).

Calcul des termes $F_a^g = P(\Phi_g \mid g = a)$

$$\begin{aligned} F_a^g &= 1_{\{g=a\}} \text{ si g est une feuille} \\ F_a^g &= [P(g \equiv g_0)F_a^{g0} + (1 - P(g \equiv g_0) \sum_b \alpha(b)F_b^{g0})] \\ &\quad \times [P(g \equiv g_1)F_a^{g1} + (1 - P(g \equiv g_1) \sum_b \alpha(b)F_b^{g1})] \text{ si g est un noeud interne.} \end{aligned}$$

Les deux sous-arbres A_0 et A_1 sont considérés séparément car ils sont indépendants conditionnellement à g.

Calcul des termes $\Pi_a^g = P(g = a \mid \Phi)$ Ce calcul nécessite l'introduction de nouvelles notations, il est explicité en annexe.

6.2.7 Prise en compte directe de l'arbre phylogénétique dans la prédiction par forward/backward

Dans les précédentes approches, les séquences sont prédites séparément, puis la prédiction consensus est réalisée à l'aide des pondérations. Il est possible de prendre en compte l'ensemble des familles reliées par un arbre phylogénétique directement dans la procédure de prédiction avec le HMM. Cette méthode sera nommée *méthode directe*.

Pour cela, les termes $b_u(x_t)$, probabilité d'émission de x_t dans l'état caché u, sont remplacés par des termes relatifs à la famille de séquences, $b_u(famille_t \mid T)$, dans les équations de l'algorithme *forward/backward*.

$b_u(famille_t \mid T)$ désigne la probabilité d'observer l'ensemble des sites de la famille reliés par l'arbre phylogénétique T. Le modèle d'évolution est celui présenté précédemment. Le calcul des $b_u(famille_t \mid T)$ est très proche des calculs d'information.

Cette démarche est analogue à celle utilisée dans la méthode PASSML [121], à la différence que nous considérons la phylogénie comme connue.

Calcul des $b_u(famille_t \mid T)$

Le modèle M1M0 suppose l'indépendance des sites. Le calcul des probabilités associées à un ensemble de séquences reliées par un arbre phylogénétique est décomposé en utilisant l'algorithme de Felsenstein [64]. Le principe de cette décomposition repose sur le fait que les probabilités associées à deux noeuds frères peuvent être calculées séparément, puisque les évènements sont indépendants conditionnellement au noeud père. Les lettres portées par les noeuds internes n'étant pas connues, la somme est réalisée sur toutes les lettres possibles.

Pour chaque acide aminé a, et chaque noeud de l'arbre, on calcule la probabilité d'observer les feuilles du sous-arbre A_g, sachant que le noeud g porte la lettre a. Pour une feuille :

$$P(\Phi_g \mid g = a) = 1_{\{g=a\}}$$

Pour un noeud interne de l'arbre :

$$\begin{aligned} P(\Phi_g \mid g = a) &= \left(\sum_{x \in \mathcal{X}} P(g_0, x) \left[(1 - P(g \equiv g_o)) b_u(x) + 1_{\{a_0 = x\}} P(g \equiv g_o) \right] \right) \\ &\times \left(\sum_{y \in \mathcal{X}} P(g_1, y) [(1 - P(g \equiv g_1)) b_u(y) + 1_{\{a_0 = y\}} P(g \equiv g_1)] \right) \end{aligned}$$

Les évènements des sous-arbres gauches et droits sont indépendants, d'où le produit des deux termes. On considère toutes les lettres possibles pour le noeud fils g_0 (somme sur les x). Si les lettres sont différentes, il y a eu mutation, ce qui arrive avec la probabilité $1 - P(g \equiv g_o)$. Cette probabilité est donnée par la longueur de la branche reliant g à g_0. La nouvelle lettre est alors tirée selon la loi de l'état caché u, sans mémoire de la lettre de départ. Si les lettres sont identiques, elles peuvent être identiques par copie (probabilité $P(g \equiv g_o)$, calculée d'après la longueur de la branche de l'arbre), ou être le résultat d'une mutation invisible (mutation avec la probabilité $1 - P(g \equiv g_1)$, puis tirage de la nouvelle lettre identique à l'ancienne).

Finalement

$$b_u(famille_t \mid T) = \sum_{a \in \mathcal{X}} b_u(a) P(\Phi \mid g = a)$$

Les séquences portant des gaps ne sont pas incluses dans ce calcul.

6.2.8 Simulation de données à l'aide d'un HMM et d'un arbre

Les résultats obtenus sur séquences réelles nous on conduit à vouloir tester les différentes méthodes sur des séquences simulées, pour, d'une part, vérifier l'implémentation de notre méthode directe, et, d'autre part, comprendre pourquoi les résultats sur séquences réels sont décevants.

Si les hypothèses de la méthode de prise en compte de l'arbre phylogénétique que nous proposons sont respectées :

1. les séquences de protéines sont générées par un modèle HMM,
2. l'évolution des séquences suit un processus de saut renouvelant, à sites indépendants,
3. les séquences homologues correspondent au même processus caché,
4. l'alignement multiple des séquences respecte le processus caché.

Il y a peu de chances pour que les séquences de protéines réelles respectent ces contraintes. Cependant, il est difficile de préjuger de l'effet de la violation de ces contraintes sur les performances de prédiction.

La simulation d'une famille de séquences se fait comme suit :

- simulation d'une séquence cachée par le HMM,
- simulation d'une séquence de protéine *ancestrale* conditionnellement à la séquence cachée,
- simulation des séquences filles le long des branches de l'arbre.

Les longueurs de branches étant exprimées en nombre de mutations pour 100 résidus, il suffit de tirer la mutation d'après cette longueur. Si une mutation à lieu, on tire un acide aminé dans la loi d'émission de l'état caché correspondant.

6.3 Résultats

6.3.1 Résultats sur séquences réelles

Comparaison des différentes méthodes pour la prédiction des structures secondaires

Nous avons testé différents schémas de pondération et la prise en compte directe de l'arbre phylogénétique pour améliorer les prédictions du modèle à 36 états cachés présenté dans le chapitre 4. Les scores Q_3 obtenus avec les différentes méthodes de prise en compte des homologues sont présentés dans le tableau 6.1. Les résultats étant très similaires sur les partitions de l'ensemble de validation croisée et sur l'ensemble de test indépendant, seuls les scores de l'ensemble de test sont indiqués.

TAB. 6.1 – Scores de prédiction obtenus par différentes méthodes de prise en compte des homologues

Méthode	Score Q_3 sur les séquences de test
Pondération de Henikoff (section 6.2.3)	75.1 %
Pondération de Thompson (section 6.2.4)	75.1 %
Pondération équitable (section 6.2.6)[a]	74.8%
Pondération équitable [b]	74.8 %
Méthode directe (section 6.2.7)	65.3%

[a]Pondérations calculées en utilisant une seule loi de renouvellement générique

[b]Trois jeux de pondérations calculés en utilisant des lois de renouvellement spécifiques des structures secondaires. Les probabilités *a posteriori* sont combinées en utilisant les pondérations correspondant à la structure considérée

La prise en compte des séquences homologues permet d'améliorer significativement la prédiction. Le score Q_3 est de 68% sur les séquences uniques, il culmine à 75 % en utilisant les homologues, soit un gain de 7 points. Les différents systèmes de pondération conduisent à des résultats similaires : entre 74.5% avec les pondérations équitables et 75.1% avec les pondérations de Henikoff ou de Thompson.

En revanche, la prise en compte directe de l'arbre phylogénétique détériore la prédic-

tion. Cette méthode de prédiction est assez proche de la démarche utilisée par Goldman et al dans leur méthode PASSML [121]. Cependant, la méthode PASSML estime dans un premier temps la phylogénie, en utilisant des taux de mutation spécifiques de chaque classe de structure secondaire (voir également [53]). Dans notre méthode nous considérons la phylogénie comme connue. D'autre part, le modèle HMM utilisé par la méthode PASSML comptaient 38 états cachés de 8 types distincts (voir chapitre 3). Leur méthode est donc plus complète (elle estime la phylogénie), mais le modèle utilisé est moins fin (de nombreuses contraintes sont introduites sur les paramètres du HMM). Les performances de prédiction avec leur modèle 38 états n'ont malheureusement pas été évaluées. Les quelques exemples présentés, avec un modèle 3 états [77], montraient que la prise en compte d'homologues proches amélioraient la prédiction, d'autant plus que les séquences prises en compte étaient proches.

Le résultat inattendu de la prédiction par notre méthode directe a motivé l'étude sur séquences simulées, présentée en fin de chapitre.

Comparaison détaillée entre la combinaison des prédictions avec Henikoff et PSIPRED

Les familles de séquences utilisées sont filtrées pour conserver les séquences les plus proches, afin de prévenir les erreurs dans les alignements multiples et les arbres résultants. Cependant, ce filtrage fait perdre une partie de l'information. En effet, si les familles entières sont utilisées pour la prédiction avec pondérations de Henikoff, le score Q_3 obtenu est de 75.9%. Les autres méthodes n'ont pas été testées sur les familles entières, en raison du temps de calcul requis. Le tableau 6.2 résume les résultats de prédiction obtenus sur l'ensemble de test indépendant par notre modèle à 36 états cachés, en combinant les prédictions avec une pondération de Henikoff et ainsi que les résultats de prédiction obtenus par la méthode PSIPRED sur les mêmes données. Le score Q_3 obtenu avec PSIPRED est proche de 79%, soit 3 points de plus qu'avec notre méthode. Ces indices montrent que la différence entre PSIPRED réside dans la prédiction des brins β : PSIPRED arrive fournit une bien meilleure sensibilité de prédiction des brins, avec très peu de perte de spécificité. Signalons toutefois que PSI-PRED est ici utilisé sans validation croisée, c'est à dire que notre ensemble de test contient éventuellement des séquences homologues du jeu d'apprentissage de cette méthode. La prise en compte des séquences homologues

dans la prédiction améliore tous les indices de performances par rapport aux prédictions uniséquences (voir table 4.9).

TAB. 6.2 – Performances comparées du modèle optimal utilisé avec les pondérations de Henikoff et de PSIPRED, sur les séquences de test

Performances du modèle HMM 36 états + pondération de Henikoff

	Prédiction par classe				Prédiction globale	
	Sensibilité	Spécificité	CCM	Score SOV	Score Q_3	Score SOV
Hélice	82.3%	82.4%	0.72	82.4%	75.9%	73.66%
Brin	56.3%	80.6%	0.60	68.5%		
Coil	80.99%	68.9%	0.60	68.6%		

Performances de PSIPRED

	Prédiction par classe				Prédiction globale	
	Sensibilité	Spécificité	CCM	Score SOV	Score Q_3	Score SOV
Hélice	82.1%	86.1%	0.75	88.5%	78.6%	77.0%
Brin	75.6%	76.8%	0.70	83.2%		
Coil	76.9%	72.9%	0.61	64.8%		

Prédiction des zones d'angles avec la combinaison de Henikoff

La prédiction des structures secondaires ne donnant aucun indice sur la structure des résidus en coil, la prédiction des angles dièdres est particulièrement précieuse pour compléter la prédiction locale. Les zones d'angles ont été prédites en utilisant le modèle à 65 états cachés présenté dans le chapitre 3, et la combinaison par les pondérations de Henikoff. Les familles de séquences non filtrées ont été utilisées. Les résultats détaillés de la prédiction figurent dans la tableau 6.3.

Globalement, cette méthode de prédiction permet de prédire la bonne zone d'angle pour 78 % des résidus. Les indices de prédiction par classe montrent que la zone g (correspondant à des angles Φ positifs) est la plus difficile à prédire : trop peu de résidus sont prédits dans cette zone. La comparaison avec les prédictions uniséquences avec le même modèle (voir table 4.11) montre que l'utilisation des homologues permet une prédiction plus spécifique mais beaucoup moins sensible de la zone d'angle g.

TAB. 6.3 – Performances du modèle à 65 états cachés + pondérations de Henikoff pour la prédiction des 3 zones d'angles dièdres

Prédiction par classe

	Sensibilité	Spécificité	CCM
Zone h	83.3%	79.7%	0.65
Zone e	78.3%	76.8%	0.63
Zone g	31.5%	69.8%	0.45

Prédiction globale

Score Q_3
78.1%

La méthode HMMSTR [36] donne un taux de bonne prédiction de 74.5% pour 3 grandes zones d'angles très similaires à celles que nous utilisons, la zone g étant également sous-prédite (sensibilité : 28%). Les résultats publiés par Yang et al [111] donnent un score Q_3 de 78.7% en utilisant les SVM et PSIPRED (là aussi, sans validation croisée pour PSIPRED). Les indices de prédiction sont comparables à ceux de notre méthode, pour les zones h et e. Pour la zone g, la prédiction de Yang est plus sensible, mais moins spécifique que la notre.

6.3.2 Résultats sur séquences simulées

Pour tester les différentes méthodes de prédiction dans un cas *idéal*, et comprendre les résultats obtenus sur séquences réelles, des familles de séquences ont été simulées en utilisant les modèles M1M0 présentés dans le chapitre 4 :

- HMM à 3 états cachés (un état caché par structure secondaire),
- HMM optimal à 36 états cachés (15 états cachés pour les hélices, 9 pour les brins 12 pour le coil),
- HMM à 75 états cachés (25 états cachés par classe de structure secondaire).

Simulation avec des arbres à 2 feuilles

Pour évaluer l'apport de séquences distantes en prédiction, nous avons simulé des familles de deux séquences, séparées de la racine par des branches de longueur variable.

500 paires de séquences de 200 acides aminés ont été simulées sous différents modèles. Les prédictions sont réalisées en utilisant les pondérations ou bien par la méthode de prise en compte directe de l'arbre phylogénétique.

L'évolution des scores de prédiction en fonction de la longueur des branches est pré-

sentée dans la figure 6.4. Sur une famille constituée de 2 séquences, le poids de chaque séquence est toujours 0.5.

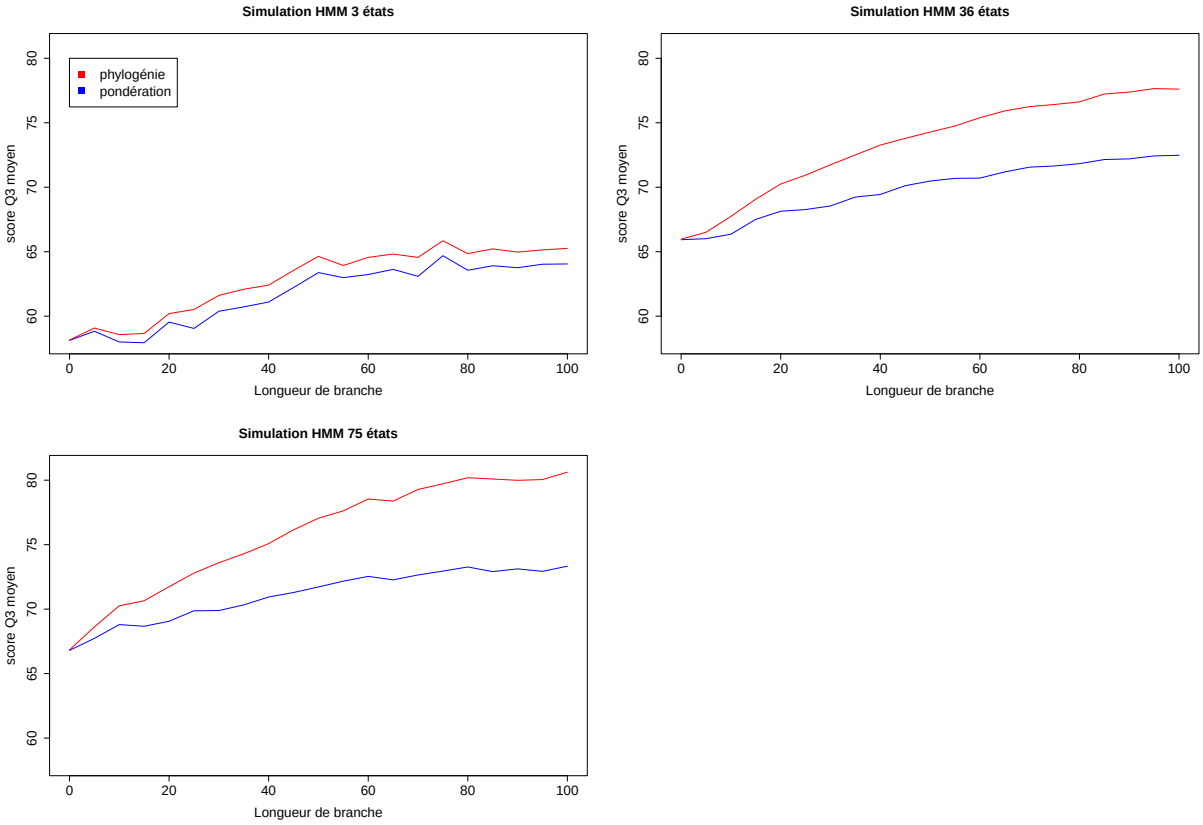

FIG. 6.4 – Évolution du score Q_3 en fonction de la longueur des branches, sur des familles de deux séquences simulées et prédites par différents modèles. En bleu, les prédictions sont réalisées en combinant les prédictions par l'algorithme forward/backward sur séquences individuelles. En rouge, la prédiction est réalisée avec prise en compte directe de l'arbre phylogénétique dans la prédiction par l'algorithme forward/backward.

Les taux de prédiction augmentent avec la longueur des branches. Ce résultat n'est pas surprenant ; il illustre le fait que des séquences distantes apportent plus d'information que des séquences proches. Les modèles plus complexes permettent de meilleures prédictions. En effet, un modèle plus riche augmente le contraste entre états cachés, ce qui permet une meilleure prédiction par rapport à un modèle simple qui simule des comportements plus globaux. Enfin, sur des séquences simulées, la méthode directe est toujours plus performante que l'utilisation des pondérations. Ceci valide notre méthode basée sur la phylogénie. L'insuccès sur les séquences réelles est probablement le fait des hypothèses trop fortes que nous avons supposées respectées.

Simulation d'après des arbres phylogénétiques réels

Nous avons simulé des familles de séquences d'après des arbres phylogénétiques réels, obtenus sur nos données. Pour cela, les familles de l'ensemble de test indépendant ont été filtrées afin de ne conserver que les séquences ayant au moins 60% d'identité de séquence avec la séquence requête, sur au moins 80% de sa longueur. 50 petites familles ont été conservées ; elles sont constituées de 2 à 10 séquences.

La simulation de familles de séquences, de longueur 200 acides aminés, a été conduite à partir des ces 50 arbres et du HMM à 36 états cachés. En parallèle, la prédiction a été menée sur les séquences réelles associées à ces arbres, avec le modèle à 36 états cachés. Le filtrage des familles a pour but de limiter les éventuelles erreurs d'alignement, puisque les séquences considérées sont relativement proches.

Les résultats de ce calcul sont portés dans le tableau 6.4.

TAB. 6.4 – Résultats de prédiction à partir de 50 arbres phylogénétiques réels

Utilisation du modèle 36 états cachés

	Séquences simulées	Séquences réelles
Prédiction par pondération (Henikoff)	Q_3 =73.8%	Q_3 =73.4%
Prédiction par la méthode directe	Q_3 =83.5%	Q_3 =68.7%

Utilisation du modèle 3 états cachés

	Séquences simulées	Séquences réelles
Prédiction par pondération (Henikoff)	Q_3 =65.8%	Q_3 =62.4%
Prédiction par la méthode directe	Q_3 =71.7%	Q_3 =62.5%

Les résultats sur séquences simulées sous les deux modèles montrent un net avantage pour la méthode de prise en compte directe de la phylogénie. Avec le modèle à 36 états cachés, le $Q3$ dépasse 80%, contre moins de 75% en utilisant la pondération des prédictions. Avec le modèle 3 états, la méthode directe arrive encore en tête : 71.7% contre 65.8% avec la pondération. En revanche, sur les séquences réelles, nous retrouvons des performances supérieures avec la pondération lors de la prédiction avec le modèle 36 états. En utilisant le modèle 3 états, les performances avec la pondération et la méthode directe sont équivalentes.

La méthode directe suppose que la séquence des états cachés est la même pour toutes les séquences d'une famille. Sur les séquences réelles, un premier écueil est la présence éventuelle d'erreurs dans l'alignement. Si la séquence cachée est effectivement commune à toutes les séquences, des erreurs d'alignement provoquent des décalages dans l'alignement de la séquence cachée commune et mettent en défaut la méthode. Ces erreurs d'alignement ont été réduites, dans une certaine mesure, par le filtrage des familles pour conserver les séquences les plus proches de la requête. Une deuxième violation des hypothèses peut provenir de l'absence d'une séquence cachée unique et commune à toutes les séquences observées d'une famille. En effet, l'hypothèse de conservation des structures secondaires dans une famille est globalement respectée, c'est même le fondement de l'utilisation des familles de séquences pour la prédiction de structure secondaire. En revanche, lors de l'utilisation d'un modèle HMM dans lequel un ensemble d'états cachés modélise la même classe structurale (cas du modèle 36 états), la même structure secondaire peut être obtenue par différentes séquences d'états cachés. Le défaut de la méthode utilisant la phylogénie avec un modèle à 36 états cachés pourrait indiquer que les séquences d'une même famille ne partagent pas la même séquence cachée.

Pour tester cette hypothèse, nous livrons ici les résultats d'une dernière simulation. En reprenant les 50 arbres phylogénétiques des petites familles de séquences, des séquences ont été simulées et prédites avec différents modèles : modèle à 3 états cachés, modèle à 36 états cachés, modèle à 75 états cachés. Les résultats des prédictions par la pondération de Henikoff et par la méthode directe sont rapportés dans le tableau 6.5.

TAB. 6.5 – Scores Q_3 obtenus sur 50 familles de séquences simulées et prédites par différents modèles

		Modèles utilisés pour **simuler** les séquences		
Modèles utilisés en **prédiction**		HMM 3 états	HMM 36 états	HMM 75 états
HMM 3 états	pondération	65.8%	64.3%	63.4%
	directe	**71.7%**	**69.0%**	**66.2%**
HMM 36 états	pondération	**59.5%**	73.8%	72.3%
	directe	58.72%	**83.5%**	**75.9%**
HMM 75 états	pondération	**59.1%**	**72.6%**	73.7%
	directe	49.9%	69.6%	**87.0%**

Lors de la prédiction sur ces séquences simulées, la méthode directe donne de meilleurs résultats que la pondération quand les séquences sont prédites avec un modèle identique ou moins complexe que le modèle ayant généré les séquences. Au contraire, quand le modèle utilisé pour la prédiction est plus complexe que le modèle utilisé en simulation, la méthode directe est systématiquement en défaut.

Des séquences simulées avec un modèle 3 états n'ont en commun que leur structure secondaire. Quand ces séquences sont analysées avec un modèle plus complexe, l'hypothèse d'un chemin caché commun n'est plus respectée au sein d'une macroclasse modélisée par plusieurs états cachés. La méthode directe est alors moins performante que la méthode des pondérations qui est une approche plus globale. Ceci est d'autant plus marqué que les modèles utilisés pour la simulation et le test sont différents : sur des séquences simulées avec un modèle à 3 états cachés la méthode de pondération donne un score Q_3 supérieur de 10 points à la méthode utilisant la phylogénie. Des simulations sur des jeux de données plus importants confirment ce comportement (voir tableau 6.6).

La méthode PASSML utilise elle aussi un couplage entre la phylogénie et un HMM. Les résultats sur des modèles 3 états ont montré l'intérêt d'utiliser les séquences homologues : la prédiction est meilleure que celle obtenue à partir d'une séquence unique ; et est meilleure qu'en utilisant une approche qui ne tient pas compte des corrélations entre séquences [77]. Malheureusement, l'apport de la phylogénie n'a pas été évalué pour le modèle à 38 états cachés. Nos résultats sur données réelles démontrent également l'apport de la prise en compte directe de la phylogénie couplée à un modèle à 3 états. Cependant, notons que l'approche par pondération donne des résultats équivalents dans ce cas.

TAB. 6.6 – Scores Q_3 obtenus sur 300 familles de séquences simulées et prédites par différents modèles

		Modèles utilisés pour **simuler** les séquences	
Modèles utilisés en **prédiction**		HMM 3 états	HMM 36 états
HMM 3 états	pondération	70.1%	67.3%
	phylogénie	**79.2%**	**69.0%**
HMM 36 états	pondération	**65.2%**	79.8%
	phylogénie	63.0%	**91.47%**

6.4 Conclusion

L'intégration de l'information des séquences homologues par le biais d'une méthode de pondération simple des prédictions réalisées indépendamment avec le modèle optimisé sur les séquences seules permet d'améliorer significativement le taux de prédiction des structures secondaires. Le score Q_3 obtenu avec le modèle optimal à 36 états cachés était de 68.1% sur les séquences seules. Il est proche de 76% avec les familles. Le défaut de notre méthode sur la prédiction des brins β persiste malgré l'utilisation des séquences homologues. La même méthodologie appliquée à la prédiction des zones d'angles permet de fournir une prédiction correcte pour 78% des résidus.

La méthode que nous proposons pour tenir compte de la phylogénie n'est pas applicable avec notre modèle à 36 états cachés. Les expériences menées sur séquences simulées montrent que cet insuccès est probablement dû à la violation d'une hypothèse forte à la base de cette méthode : la conservation d'un processus caché commun à une famille de séquences alignées. Sur des séquences réelles, la méthode des pondérations est donc plus robuste.

Le calcul de l'information conditionnelle portée par un arbre phylogénétique fournit des comptages corrigés d'acides aminés. Ces comptages pourront être utilisés pour estimer les paramètres du HMM directement sur des familles de séquences.

Conclusion et perspectives

Le principal objectif de cette thèse était de mettre au point une méthode de prédiction de la structure locale des protéines à l'aide des modèles de chaînes de Markov cachées.

En amont de la prédiction, nous avons développé une nouvelle méthode d'assignation automatique des structures secondaires, dénommée KAKSI. La comparaison détaillée des résultats de KAKSI avec ceux de différentes autres méthode disponibles, ainsi que l'analyse de la géométrie des hélices α définies par KAKSI a montré que les assignations produites sont satisfaisantes à plusieurs niveaux. Notamment, les hélices α assignées par KAKSI sont plus régulières que celles assignées par STRIDE, sans pour autant que notre méthode en assigne moins. La détection des coudes dans les hélices permet de proposer des assignations plus pertinentes que STRIDE dans certains cas difficiles.

En utilisant les assignations fournies par KAKSI, nous avons ensuite essayé plusieurs types de modèles HMM pour la prédiction de structure locale. Les apports des différents modèles proposés successivement dans cette thèse pour la prédiction des structures secondaires sont rapportés dans le tableau 6.7, pour illustrer la discussion.

TAB. 6.7 – Récapitulatif des performances de prédiction de structure secondaire par les différents modèles

Modèle	Score Q_3
M1M0 à 3 états cachés	58.2%
PM ordre 2 à 3 états cachés	61.1%
M1M0 à 21 états cachés	65.7%
M1M0 à 36 états cachés	68.1%
M1M0 à 36 états cachés + Henikoff	75.9%

L'utilisation des modèles de chaînes de Markov cachées était motivée par la souplesse de modélisation et le fait que dans ces modèles, les caractéristiques prises en compte par le modèle sont explicites. La modélisation basique des structures secondaires par un modèle à 3 états cachés s'est vite heurtée au problème de sur-apprentissage lié à l'utilisation d'ordres élevés sur les séquences de protéines. Néanmoins, les divers schémas testés pour la prise en compte de la mémoire de la séquence, notamment les modèles de Markov parcimonieux, ont montré la nécessité d'utiliser des schémas de dépendances complexes et spécifiques par classe. Les arbres de suffixes obtenus par modèles parcimonieux offrent une interprétation intuitive des dépendances entre sites, et les taux de prédictions obtenus semblent encourageants (Q_3 d'environ 61 %). Il semble intéressant de poursuivre dans cette

voie, en particulier pour choisir les partitions sans spécifier de groupes.

La proposition d'un modèle de type M1M0 à 21 états cachés, construit sur des bases biologiques, a permis d'améliorer significativement les performances. Les contraintes de modélisation dans ce cadre sont très fortes. Les choix de modélisation sont cruciaux, ce qui est délicat lorsqu'on dispose de peu d'informations *a priori*, comme c'est le cas pour les brins. Une analyse statistique préliminaire des séquences à modéliser, en terme de mots exceptionnels, a permis de guider ces choix.

A l'inverse d'une démarche de modélisation guidée par des *a priori* sur les séquences, des modèles peu contraints ont été estimés. La question qui se pose alors est celle du choix de modèle. Nous avons montré que les performances prédictives et les critères statistiques s'accordent à sélectionner des modèles de taille relativement limitée. Un modèle optimal, à 36 états cachés, a ainsi été retenu. Outre qu'il permet d'améliorer la prédiction par rapport au modèle à 21 états cachés (Q_3=68%), l'obtention même du modèle est un résultat intéressant. Les *a priori* de modélisation étant extrêmement limités, un tel modèle est informatif en soit car il révèle les caractéristiques prises en compte par le modèle. La structure finale du modèle optimal montre une organisation interne des structures secondaires bien plus complexe que les propositions que nous avions pu faire. Certaines caractéristiques du modèle peuvent être directement reliées à des précédentes études statistiques des protéines, comme la dépendance à la position dans les hélices.

Les méthodes de prédiction de structure secondaire laissent non décrits une forte proportion des résidus, prédits en apériodique. Il nous a donc semblé crucial d'envisager une prédiction de structure locale qui permette de décrire les résidus en coil. L'utilisation des HMM pour la prédiction des angles dièdres Φ/Ψ permet de prédire la bonne zone d'angle pour environ 73% des résidus, en particulier le taux de bonne prédiction est d'environ 61% pour l'apériodique.

L'intégration des séquences homologues pour améliorer la prédiction a été envisagée en utilisant deux méthodes : une combinaison après coup des prédictions réalisées sur séquences indépendantes et un couplage direct de la phylogénie avec le HMM. Dans ce cadre, nous avons proposé un calcul d'information conditionnelle d'un arbre phylogénétique (ou comptage corrigé), permettant de dériver un système de pondération tenant compte de manière stricte de l'arbre phylogénétique et des séquences portées par l'arbre. Ce système de pondération s'est avéré équivalent à des pondérations plus classiques. Cependant, la

possibilité de changer la loi de renouvellement ouvre la voie à des méthodes plus fines de combinaisons, dont certaines restent à explorer. Les résultats obtenus sur séquences réelles et simulées ont montré que les séquences homologues ne partagent probablement pas la même séquence cachée dans notre modèle à 36 états cachés. En d'autre termes, ce modèle, s'il est optimal pour la prédiction sur séquences seules, capture des caractéristiques qui ne sont pas conservées dans une famille. En conséquence, le couplage direct du HMM avec la phylogénie est mis en défaut lorsqu'il est utilisé avec des modèles complexes qui ne garantissent pas la conservation du chemin caché. Cependant, le calcul de l'information mutuelle dans les arbres phylogénétiques rend possible l'estimation directe de modèles sur les données phylogénétiques. Un tel modèle capturerait nécessairement des caractéristiques conservées par l'évolution. Il serait intéressant d'estimer ainsi des modèles pour ré-évaluer l'apport du couplage direct.

La prédiction des zones d'angles en utilisant les pondérations de Henikoff donne en taux de 78% de résidus correctement prédits, avec une bonne prédiction de 62% dans l'apériodique.

La prédiction locale ainsi réalisée ouvre la voie vers le développement de stratégies de prédiction *de novo* qui permettront de proposer des modèles globaux dans les cas de prédiction difficiles. La prédiction locale peut également être utilisée pour améliorer la prédiction des boucles correspondant à des insertions dans les modèles prédits par homologie ou reconnaissance de repliement. La prédiction de structure *de novo* peut, dans certains cas, aider à la prédiction de fonction. Une perspective à plus long terme est donc l'aide à l'annotation de génomes.

Bibliographie

[1] R. Adamczak, A. Porollo, and J. Meller. Combining prediction of secondary structure and solvent accessibility in proteins. *Proteins*, 59(3) :467–475, 2005.

[2] V. Ahola, T. Aittokallio, E. Uusipaikka, and M. Vihinen. Efficient estimation of emission probabilities in profile hidden markov models. *Bioinformatics*, 19(18) :2359–68, 2003.

[3] P. Aloy, A. Stark, C. Hadley, and R. B. Russell. Predictions without templates : new folds, secondary structure, and contacts in casp5. *Proteins*, 53 Suppl 6 :436–456, 2003.

[4] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped blast and psi-blast : a new generation of protein database search programs. *Nucleic Acids Res.*, 25(17) :3389–3402, 1997.

[5] C. Andersen and B. Rost. *Structural Bioinformatics*, chapter Automated Secondary Structure Assignment. Wiley, 2003.

[6] K. Asai, S. Hayamizu, and K. Handa. Prediction of protein secondary structure by the hidden markov model. *Comput. Appl. Biosci.*, 9(2) :141–146, 1993.

[7] M. Asogawa. Beta-sheet prediction using inter-strand residue pairs and refinement with hopfield neural network. In *Proc Int Conf Intell Syst Mol Biol*, volume 5, pages 48–51, 1997.

[8] R. Aurora and G.D. Rose. Helix capping. *Protein Sci*, 7(1) :21–38, 1998.

[9] R. Aurora, R. Srinivasan, and G.D. Rose. Rules for alpha-helix termination by glycine. *Science*, 264(5162) :1126–30, 1994.

[10] Y. Aydin, Z. Altunbasak and M. Borodovsky. Protein secondary structure prediction with semi-markov hmms. In *IEEE International Conference on Acoustics Speech and Signal Processing*, 2004.

[11] D. Baker and A. Sali. Protein structure prediction and structural genomics. *Science*, 294(5540) :93–96, 2001.

[12] P. Baldi, S. Brunak, P. Frasconi, G. Soda, and G. Pollastri. Exploiting the past and the future in protein secondary structureprediction. *Bioinformatics*, 15(11) :937–946, 1999.

[13] P. Baldi, G. Pollastri, CA. Andersen, and S. Brunak. Matching protein beta-sheet partners by feedforward and recurrent neural networks. In *Proc Int Conf Intell Syst Mol Biol*, number 8, pages 25–36, 2000.

[14] M. Bansal, S. Kumar, and R. Velavan. Helanal : a program to characterize helix geometry in proteins. *J Biomol Struct Dyn*, 17(5) :811–9, 2000.

[15] D.J. Barlow and JM. Thornton. Helix geometry in proteins. *J Mol Biol*, 201(3) :601–19, 1988.

[16] C. Barrett, R. Hughey, and K. Karplus. Scoring hidden markov models. *Comput. Appl. Biosci.*, 13(2) :191–199, 1997.

[17] A. Bateman, L. Coin, R. Durbin, R. D. Finn, V. Hollich, S. Griffiths-Jones, A. Khanna, M. Marshall, Si. Moxon, E. L. L. Sonnhammer, D. J. Studholme, C. Yeats, and S. R. Eddy. The pfam protein families database. *Nucleic Acids Res.*, 32(Database issue) :D138–141, 2004.

[18] A. Berchtold. Estimation in the mixture transition distribution model. *Journal of Time Series Analysis*, 22(4) :379–397, 2001.

[19] A. Berchtold and A.E. Raftery. The mixture transition distribution model for high-order markov chains and non-gaussian time series. *Statistical Science*, 17(3) :328–356, 2002.

[20] H.J.C. Berendsen, D. van der Spoel, and R. van Drunen. Gromacs : A message-passing parallel molecular dynamics implementation. *Comp. Phys. Comm.*, 91 :43–56, 1995.

[21] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic Acids Res*, 28(1) :235–242, 2000.

[22] V. Biou, J. F. Gibrat, J. M. Levin, B. Robson, and J. Garnier. Secondary structure prediction : combination of three different methods. *Protein Eng.*, 2(3) :185–191, 1988.

[23] I. Bonneau, R. Ruczinski and D. Tsai, J. Baker. Contact order and ab initio protein structure prediction. *Protein Sci*, 11(8) :1937–44, 2002.

[24] R. Bonneau, J. Tsai, I. Ruczinski, D. Chivian, C. Rohl, C.E. Strauss, and D. Baker. Rosetta in casp4 : progress in ab initio protein structure prediction. *Proteins*, (Suppl 5) :S119–26, 2001.

[25] P.-Y. Bourguignon and D. Robelin. Modèles de markov parcimonieux : sélection de modèle et estimation. In *Actes de JOBIM*, 2004.

[26] P. E. Bourne. *Structural Bioinformatics*, chapter CASP and CAFASP experiments and their findings. Wiley, 2003.

[27] P. E. Bourne and H. Weissig, editors. *Structural Bioinformatics*, volume 44 of *Methods of biochemical analysis*. Willey-Liss, 2003.

[28] A. Bouvier, F. Gélis, and S. Schbath. *RMES : Programs to Find Words with Unexpected Frequencies in DNA Sequences, User Guide (in french)*, 1999.

[29] P. M. Bowers, C. E. Strauss, and D. Baker. De novo protein structure determination using sparse nmr data. *J. Biomol NMR*, 18(4) :311–318, 2000.

[30] P. Bradley, D. Chivian, J. Meiler, K.M. Misura, C.A. Rohl, W. Schief, W.J. R. Wedemeyer, O. Schueler-Furman, P. Murphy, J. Schonbrun, C.E. Strauss, and D. Baker. Rosetta predictions in casp5 : successes, failures, and prospects for complete automation. *Proteins*, 53(Suppl 6) :457–68, 2003.

[31] S.E. Brenner, P. Koehl, and M. Levitt. The astral compendium for protein structure and sequence analysis. *Nucleic Acids Res*, 28(1) :254–6, 2000.

[32] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus. Charmm : A program for macromolecular energy, minimization, and dynamics calculations. *J. Comp. Chem.*, 4 :187–217, 1983.

[33] M. Brown, R. Hughey, A. Krogh, I. S. Mian, K. Sjolander, and D. Haussler. Using dirichlet mixture priors to derive hidden markov models for protein families. *Proc Int Conf Intell Syst Mol. Biol.*, 1 :47–55, 1993.

[34] C. Bystroff and D. Baker. Prediction of local structure in proteins using a library of sequence-structure motifs. *J Mol Biol*, 281(3) :565–77, 1998.

[35] C. Bystroff and Y. Shao. Fully automated ab initio protein structure prediction using i-sites, hmmstr and rosetta. *Bioinformatics*, pages S54–61, 2002.

[36] C. Bystroff, V. Thorsson, and D. Baker. Hmmstr : a hidden markov model for local sequence-structure correlations in proteins. *J Mol Biol*, 301(1) :173–90, 2000.

[37] P. Bühlmann and A.J. Wyner. Variable length markov chains. *Annals of Statistics*, 27(2) :480–513, 1999.

[38] A.C. Camproux, R. Gautier, and P. Tuffery. A hidden markov model derived structural alphabet for proteins. *J Mol Biol*, 339(3) :591–605, 2004.

[39] J.-P. Cartailler and H. Luecke. Structural and functional characterization of pi bulges and other short intrahelical deformations. *Structure (Camb)*, 12(1) :133–144, 2004.

[40] A.W. Chan, E.G. Hutchinson, D. Harris, and J.M. Thornton. Identification, classification, and analysis of beta-bulges in proteins. *Protein Sci*, 2(10) :1574–90, 1993.

[41] P. Y. Chou and G. D. Fasman. Conformational parameters for amino acids in helical, beta-sheet, and random coil regions calculated from proteins. *Biochemistry*, 13(2) :211–222, 1974.

[42] W. Chu, Z. Ghahramani, and D. L. Wild. A graphical model for protein secondary structure prediction, international conference on machine learning. In *International Conference on Machine Learning*, page 161 :168, 2004.

[43] G. A. Churchill. Stochastic models for heterogeneous dna sequences. *Bull Math Biol.*, 51(1) :79–94, 1989.

[44] N. Colloc'h, C. Etchebest, E. Thoreau, B. Henrissat, and J.P. Mornon. Comparison of three algorithms for the assignment of secondary structure in proteins : the advantages of a consensus assignment. *Protein Eng*, 6(4) :377–82, 1993.

[45] J. L. Cornette, K. B. Cease, H. Margalit, J. L. Spouge, J. A. Berzofsky, and C. DeLisi. Hydrophobicity scales and computational techniques for detecting amphipathic structures in proteins. *J. Mol. Biol.*, 195(3) :659–685, 1987.

[46] G. E. Crooks and S. E. Brenner. Protein secondary structure : entropy, correlations and prediction. *Bioinformatics*, 2004.

[47] A. Dal Palu, A. Dovier, and F. Fogolari. Constraint logic programming approach to protein structure prediction. *BMC Bioinformatics*, 5(1) :186–186, 2004.

[48] G. Dantas, B. Kuhlman, D. Callender, M. Wong, and D. Baker. A large scale test of computational protein design : folding and stability of nine completely redesigned globular proteins. *J. Mol. Biol.*, 332(2) :449–460, 2003.

[49] A. G. de Brevern, C. Etchebest, and S. Hazout. Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks. *Proteins*, 41(3) :271–287, 2000.

[50] A. L. Delcher, S. Kasif, H. R. Goldberg, and W. H. Hsu. Protein secondary structure modelling with probabilistic networks. *Proc Int Conf Intell Syst Mol. Biol.*, 1 :109–117, 1993.

[51] K. A. Dill, K. M. Fiebig, and H. S. Chan. Cooperativity in protein-folding kinetics. *Proc Natl Acad Sci. U S A*, 90(5) :1942–1946, 1993.

[52] J. Donohue. Hydrogen bonded helical configurations of the polypeptide chain. *PNAS*, 39 :470–478, 1953.

[53] L. Dudoignon, E. Remy, J.-L. Risler, and F. Campillo. Variation du taux d'évolution le long de séquences de protéines. In *Actes de JOBIM*, 2001.

[54] R. L. Jr. Dunbrack and M. Karplus. Backbone-dependent rotamer library for proteins. application to side-chain prediction. *J. Mol. Biol.*, 230(2) :543–574, 1993.

[55] F. Dupuis, J.F. Sadoc, and J.P. Mornon. Protein secondary structure assignment through voronoi tessellation. *Proteins*, 55(3) :519–28, 2004.

[56] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. *Biological sequence analysis*. Cambridge University Press, 1998.

[57] S R Eddy. Profile hidden markov models. *Bioinformatics*, 14(9) :755–763, 1998.

[58] R. C. Edgar and K. Sjolander. Coach : profile-profile alignment of protein families using hidden markov models. *Bioinformatics*, 20(8) :1309–1318, 2004.

[59] T. Edgoose and D.L. Allison, L.and Dowe. An mml classification of protein structure that knows about angles and sequence. In *Pac Symp Biocomput*, pages 585–96, 1998.

[60] D Eisenberg, R M Weiss, and T C Terwilliger. The helical hydrophobic moment : a measure of the amphiphilicity of a helix. *Nature*, 299(5881) :371–374, 1982.

[61] D. E. Engel and William F. DeGrado. Amino acid propensities are position-dependent throughout the length of alpha-helices. *J. Mol. Biol.*, 337(5) :1195–1205, 2004.

[62] E. Eskin, W. S. Noble, and Y. Singer. Protein family classification using sparse markov transducers. *J. Comput. Biol.*, 10(2) :187–213, 2003.

[63] N. Eswar, C. Ramakrishnan, and N. Srinivasan. Stranded in isolation : structural role of isolated extended strands in proteins. *Protein Eng*, 16(5) :331–9, 2003.

[64] J. Felsenstein. Evolutionary trees from dna sequences : a maximum likelihood approach. *J. Mol. Evol.*, 17(6) :368–376, 1981.

[65] A. Figureau, M. A. Soto, and J. Toha. A pentapeptide-based method for protein secondary structure prediction. *Protein Eng.*, 16(2) :103–107, 2003.

[66] D. Fischer, A. Elofsson, L. Rychlewski, F. Pazos, A. Valencia, B. Rost, A. R. Ortiz, and R. L. Jr Dunbrack. Cafasp2 : the second critical assessment of fully automated structure prediction methods. *Proteins*, Suppl 5 :171–183, 2001.

[67] M.N. Fodje and S. Al-Karadaghi. Occurrence, conformational features and amino acid propensities for the pi-helix. *Protein Eng*, 15(5) :353–8, 2002.

[68] L. Fourrier, C. Benros, and A.G. de Brevern. Use of a structural alphabet for analysis of short loops connecting repetitive structures. *BMC Bioinformatics*, 5(1) :58, 2004.

[69] D. Frishman and P. Argos. Knowledge-based protein secondary structure assignment. *Proteins*, 23(4) :566–79, 1995.

[70] D. Frishman and P. Argos. Incorporation of non-local interactions in protein secondary structure prediction from the amino acid sequence. *Protein Eng*, 9(2) :133–42, Feb 1996.

[71] J Garnier, J F Gibrat, and B Robson. Gor method for predicting protein secondary structure from amino acid sequence. *Methods Enzymol*, 266 :540–553, 1996.

[72] J. Garnier, D. J. Osguthorpe, and B. Robson. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *J. Mol. Biol.*, 120(1) :97–120, 1978.

[73] C. Geourjon and G. Deleage. Sopma : significant improvements in protein secondary structure prediction by consensus prediction from multiple alignments. *Comput. Appl. Biosci.*, 11(6) :681–684, 1995.

[74] J. F. Gibrat, J. Garnier, and B. Robson. Further developments of protein secondary structure prediction using information theory. new parameters and consideration of residue pairs. *J. Mol. Biol.*, 198(3) :425–443, 1987.

[75] J. F. Gibrat, B. Robson, and J. Garnier. Influence of the local amino acid sequence upon the zones of the torsional angles phi and psi adopted by residues in proteins. *Biochemistry*, 30(6) :1578–1586, 1991.

[76] J.F. Gibrat and S.H. Madej, T.and Bryant. Surprising similarities in structure comparison. *Curr Opin Struct Biol*, 6(3) :377–85, 1996.

[77] N. Goldman, J. L. Thorne, and D. T. Jones. Using evolutionary trees in protein secondary structure prediction and other comparative sequence analyses. *J. Mol. Biol.*, 263(2) :196–208, 1996.

[78] N. Goldman, J. L. Thorne, and D. T. Jones. Assessing the impact of secondary structure and solvent accessibility on protein evolution. *Genetics*, 149(1) :445–458, 1998.

[79] Y. Guermeur, G. Pollastri, A. Elisseeff, H. Zelus, D.and Paugam-Moisy, and P. Baldi. Combining protein secondary structure prediction models with ensemble methods of optimal complexity. *Neurocomputing*, pages 305–327, 2004.

[80] C. Hardin, T. V. Pogorelov, and Z. Luthey-Schulten. Ab initio protein structure prediction. *Curr Opin Struct Biol.*, 12(2) :176–181, 2002.

[81] S. Henikoff and JG. Henikoff. Position-based sequence weights. *J Mol Biol*, 243(4) :574–8, 1994.

[82] U. Hobohm, M. Scharf, R Schneider, and C. Sander. Selection of a representative set of structures from the brookhaven protein data bank. *Protein Science*, 1 :409–417, 1992).

[83] H-J. Hu, Y. Pan, R. Harrison, and P. C. Tai. Improved protein secondary structure prediction using support vector machine with a new encoding scheme and an advanced tertiary classifier. *IEEE Trans. Nanobioscience*, 3(4) :265–271, 2004.

[84] S. Hua and Z. Sun. A novel method of protein secondary structure prediction with high segment overlap measure : support vector machine approach. *J Mol Biol.*, 308(2) :397–407, 2001.

[85] W. Humphrey, A. Dalke, and K. Schulten. Vmd : visual molecular dynamics. *J Mol Graph*, 14(1) :33–8, 27–8, 1996.

[86] C. G. Hunter and S. Subramaniam. Protein local structure prediction from sequence. *Proteins*, 50(4) :572–579, 2003.

[87] E. G. Hutchinson and J. M ; Thornton. A revised set of potentials for beta-turn formation in proteins. *Protein Sci.*, 3(12) :2207–2216, 1994.

[88] EG. Hutchinson, RB. Sessions, JM. Thornton, and DN. Woolfson. Determinants of strand register in antiparallel beta-sheets of proteins. *Protein Sci*, 7(11) :2287–300, Nov 1998.

[89] F. Jiang. Prediction of protein secondary structure with a reliability score estimated by local sequence clustering. *Protein Eng.*, 16(9) :651–657, 2003.

[90] D.T. Jones. Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol.*, 292(2) :195–202, 1999.

[91] W. Kabsch and C. Sander. Dictionary of protein secondary structure : pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12) :2577–637, 1983.

[92] E T Kaiser and F J Kezdy. Amphiphilic secondary structure : design of peptide hormones. *Science*, 223(4633) :249–255, 1984.

[93] L. Kall, A. Krogh, and E. L. L. Sonnhammer. An hmm posterior decoder for sequence feature prediction that includes homology information. *Bioinformatics*, 21(Suppl 1) :i251–i257, 2005.

[94] M. Kanehisa. A multivariate analysis method for discriminating protein secondary structural segments. *Protein Eng*, 2(2) :87–92, 1988.

[95] R. Karchin, M. Cline, Y. Mandel-Gutfreund, and K. Karplus. Hidden markov models that use predicted local structure for fold recognition : alphabets of backbone geometry. *Proteins*, 51(4) :504–14, 2003.

[96] K. Karplus. Evaluating regularizers for estimating distributions of amino acids. *Proc Int Conf Intell Syst Mol. Biol.*, 3 :188–196, 1995.

[97] K. Karplus, C. Barrett, and R. Hughey. Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14(10) :846–856, 1998.

[98] K. Karplus, K. Sjolander, C. Barrett, M. Cline, D. Haussler, R. Hughey, L. Holm, and C. Sander. Predicting protein structure using hidden markov models. *Proteins*, Suppl 1 :134–139, 1997.

[99] S. Kawashima, H. Ogata, and M. Kanehisa. Aaindex : Amino acid index database. *Nucleic Acids Res.*, 27(1) :368–369, 1999.

[100] G. T. Kilosanidze, A. S. Kutsenko, N. G. Esipova, and V. G. Tumanyan. Analysis of forces that determine helix formation in alpha-proteins. *Protein Sci.*, 13(2) :351–357, 2004.

[101] G.T. Kilosanidze, A.S. Kutsenko, N.G. Esipova, and V.G. Tumanyan. Use of molecular mechanics for secondary structure prediction. is it possible to reveal alpha-helix ? *FEBS Lett*, 510 :13–6, 2002.

[102] D.E. Kim, D. Chivian, and D. Baker. Protein structure prediction and analysis using the robetta server. *Nucleic Acids Res*, 32(Web Server issue) :W526–31, 2004.

[103] H. Kim and H. Park. Protein secondary structure prediction based on an improved support vector machines approach. *Protein Eng.*, 16(8) :553–560, 2003.

[104] S. Kim. Protein beta-turn prediction using nearest-neighbor method. *Bioinformatics*, 20(1) :40–4, 2004.

[105] S.M. King and W.C. Johnson. Assigning secondary structure from protein coordinate data. *Proteins*, 3(35), 1999.

[106] I.Y. Koh, V.A. Eyrich, M.A. Marti-Renom, D. Przybylski, M.S. Madhusudhan, N. Eswar, O. Grana, F. Pazos, A. Valencia, A. Sali, and B. Rost. Eva : evaluation of protein structure prediction servers. *Nucleic Acids Res*, 31(13) :3311–5, Jul 2003.

[107] P. J. Kraulis. Molscript : A program to produce both detailed and schematic plots of protein structures. *J Applied Crystallogr*, 24 :946–950, 1991.

[108] A. Krogh. Hidden Markov models for labeled sequences. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition.*, pages 140–44, Los Alamitos, California, Oct 1994. IEEE Computer Society Press.

[109] A. Krogh, M. Brown, I. S. Mian, K. Sjolander, and D. Haussler. Hidden markov models in computational biology. applications to protein modeling. *J. Mol. Biol.*, 235(5) :1501–1531, 1994.

[110] A. Krogh and S.K. Riis. *Advances in Neural Information Processing Systems*, volume 8, chapter Prediction of beta sheets in proteins, pages 917–923. MIT Press, Cambridge, MA, USA, 1996.

[111] R. Kuang, C.S. Leslie, and A.S. Yang. Protein backbone angle prediction with machine learning approaches. *Bioinformatics*, 20(10) :1612–21, 2004.

[112] S. Kumar and M. Bansal. Structural and sequence characteristics of long alpha-helices in globular proteins. *Biophys Journal*, 71 :1574–86, 1996.

[113] S. Kumar and M. Bansal. Dissecting alpha-helices : position-specific analysis of alpha-helices in globular proteins. *Proteins*, 31(4) :460–476, 1998.

[114] S. Kumar and M. Bansal. Geometrical and sequence characteristics of alpha-helices in globular proteins. *Biophys J*, 75(4) :1935–44, 1998.

[115] G. Labesse, N. Colloc'h, J. Pothier, and J.P. Mornon. P-sea : a new efficient assignment of secondary structure from c alpha trace of proteins. *Comput Appl Biosci*, 13(3) :291–5, 1997.

[116] J. M. Levin. Exploring the limits of nearest neighbour secondary structure prediction. *Protein Eng.*, 10(7) :771–776, 1997.

[117] J. M. Levin, B. Robson, and J. Garnier. An algorithm for secondary structure determination in proteins based on sequence similarity. *FEBS Lett*, 205(2) :303–308, 1986.

[118] M. Levitt and J. Greer. Automatic identification of secondary structure in globular proteins. *J Mol Biol*, 114(2) :181–239, 1977.

[119] P. N. Lewis, F. A. Momany, and H. A. Scheraga. Folding of polypeptide chains in proteins : A proposed mechanism for folding. *PNAS*, 68 :2293–2297, 1971.

[120] V I Lim. Structural principles of the globular organization of protein chains. a stereochemical theory of globular protein secondary structure. *J. Mol. Biol.*, 88(4) :857–872, 1974.

[121] P. Lio, N. Goldman, J. L. Thorne, and D. T. Jones. Passml : combining evolutionary inference and protein secondary structure prediction. *Bioinformatics*, 14(8) :726–733, 1998.

[122] B.W. Low and R.B. Baybutt. The pi-helix -a hydrogen bonded configuration of the polypeptide chain. *J Am Chem Soc*, 74 :5806, 1952.

[123] R. Luethy, J. U. Bowie, and D. Eisenberg. Assessment of protein models with three-dimensional profiles. *Nature*, 356(6364) :83–85, 1992.

[124] M. Madera and J. Gough. A comparison of profile hidden markov model procedures for remote homology detection. *Nucleic Acids Res.*, 30(19) :4321–4328, 2002.

[125] Yael Mandel-Gutfreund and Lydia M Gregoret. On the significance of alternating patterns of polar and non-polar residues in beta-strands. *J. Mol. Biol.*, 323(3) :453–461, 2002.

[126] A. Marin, J. Pothier, K.Z. Zimmermann, and J.F. Gibrat. Frost : a filter-based fold recognition method. *Proteins*, 49(4) :493–509, 2002.

[127] B. W. Matthews. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochim Biophys. Acta*, 405(2) :442–451, 1975.

[128] JS. Merkel, JM. Sturtevant, and L. Regan. Sidechain interactions in parallel beta sheets : the energetics of cross-strand pairings. *Structure Fold Des*, 7(11) :1333–43, Nov 1999.

[129] C. Micheletti, F. Seno, and A. Maritan. Recurrent oligomers in proteins : an optimal scheme reconciling accurate and concise backbone representations in automated folding and design studies. *Proteins*, 40(4) :662–674, 2000.

[130] V. Miele, P.-Y. Bourguignon, D. Robelin, G. Nuel, and H. Richard. seq++ : analyzing biological sequences with a range of markov-related models. *Bioinformatics*, 21(11) :2783–2784, 2005.

[131] S. Muggleton, R.D. King, and M.J. Sternberg. Protein secondary structure prediction using logic-based machine learning. *Protein Eng*, 5(7) :647–57, 1992.

[132] F. Muri. *Comparaison d'algorithmes d'identification de chaines de Markov cachées et application à la détection de régions homogènes dans les sequences d'ADN.* PhD thesis, Université Rene Descartes, Paris 5, 1997.

[133] A.G. Murzin, S.E. Brenner, T. Hubbard, and C. Chothia. Scop : a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol*, 247(4) :536–40, 1995.

[134] P. Nicolas. *Mise au point et utilisation de modèles de chaînes de Markov cachées pour l'analyse de séquences ADN.* PhD thesis, Université d'Evry Val d'Essonne, 2003.

[135] P. Nicolas, A-S. Tocquet, and F. Muri-Majoube. *SHOW User Manual*, 2004.

[136] B. Oliva, P. A. Bates, E. Querol, F. X. Aviles, and M. J. Sternberg. An automated classification of the structure of protein loops. *J. Mol. Biol.*, 266(4) :814–830, 1997.

[137] A.D. Orengo, C.A. Michie, S. Jones, D.T. Jones, M.B. Swindells, and J.M. Thornton. Cath–a hierarchic classification of protein domain structures. *Structure*, 5(8) :1093–108, 1997.

[138] D. J. Osguthorpe. Ab initio protein folding. *Curr Opin Struct Biol.*, 10(2) :146–152, 2000.

[139] M. Ouali and R. D. King. Cascaded multiple classifiers for secondary structure prediction. *Protein Sci.*, 9(6) :1162–1176, 2000.

[140] C C Palliser, M W MacArthur, and D A Parry. Surface beta-strands in proteins : identification using an hydropathy technique. *J. Struct Biol.*, 132(1) :63–71, 2000.

[141] B. H. Park and M. Levitt. The complexity and accuracy of discrete state models of protein structure. *J. Mol. Biol.*, 249(2) :493–507, 1995.

[142] L. Pauling and R.B. Corey. The pleated sheet, a new layer configuration of polypeptide chains. *Proc Natl Acad Sci U S A*, 37(5) :251–6, 1951.

[143] L. Pauling, R.B. Corey, and H.R. Branson. The structure of proteins ; two hydrogen-bonded helical configurations of the polypeptide chain. *Proc Natl Acad Sci U S A*, 37(4) :205–11, 1951.

[144] M.F. Perutz, J.C. Kendrew, and H.C. Watson. Some relations between polypeptide chain configuration and amino acid sequence. *J. Mol. Biol.*, 13 :669–678, 1965.

[145] G. Pollastri and A. McLysaght. Porter : a new, accurate server for protein secondary structure prediction. *Bioinformatics*, 21(8) :1719–1720, 2005.

[146] G. Pollastri, D. Przybylski, B. Rost, and P. Baldi. Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. *Proteins*, 47(2) :228–35, 2002.

[147] D. Przybylski and B. Rost. Alignments grow, secondary structure prediction improves. *Proteins*, 46(2) :197–205, 2002.

[148] N. Qian and T. J. Sejnowski. Predicting the secondary structure of globular proteins using neural network models. *J. Mol. Biol.*, 202(4) :865–884, 1988.

[149] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77 :257–286, 1989.

[150] G. N. Ramachandran and V. Sasisekharan. Conformation of polypeptides and proteins. *Adv Protein Chem.*, 23 :283–438, 1968.

[151] F.M. Richards and C.E. Kundrot. Identification of structural motifs from protein coordinate data : secondary structure and first-level supersecondary structure. *Proteins*, 3(2) :71–84, 1988.

[152] J.S. Richardson, E.D. Getzoff, and D.C. Richardson. The beta bulge : a common small unit of nonrepetitive protein structure. *Proc Natl Acad Sci U S A*, 75(6) :2574–8, 1978.

[153] J.S. Richardson and D.C. Richardson. Natural beta-sheet proteins use negative design to avoid edge-to-edge aggregation. *Proc Natl Acad Sci U S A*, 99(5) :2754–9, 2002.

[154] SK. Riis and A. Krogh. Improving prediction of protein secondary structure using structured neural networks and multiple sequence alignments. *J Comput Biol*, 3(1) :163–83, 1996.

[155] B. Robson and J. Garnier. *Introduction to Proteins and Protein Engineering*. Elsevier Press, Amsterdam, 1986.

[156] B. Robson and E. Suzuki. Conformational properties of amino acid residues in globular proteins. *J. Mol. Biol.*, 107(3) :327–356, 1976.

[157] M J Rooman, J P Kocher, and S J Wodak. Prediction of protein backbone conformation based on seven structure assignments. influence of local interactions. *J. Mol. Biol.*, 221(3) :961–979, 1991.

[158] M. J. Rooman, J. Rodriguez, and S. J. Wodak. Automatic definition of recurrent local structure motifs in proteins. *J. Mol. Biol.*, 213(2) :327–336, 1990.

[159] B. Rost. Review : protein secondary structure prediction continues to rise. *J Struct Biol*, 134(2-3) :204–18, May-Jun 2001.

[160] B. Rost. Prediction in 1d : secondary structure, membrane helices, and accessibility. *Methods Biochem Anal*, 44 :559–87, 2003.

[161] B. Rost and C. Sander. Improved prediction of protein secondary structure by use of sequence profiles and neural networks. *Proc Natl Acad Sci U S A*, 90(16) :7558–62, 1993.

[162] B. Rost and C. Sander. Combining evolutionary information and neural networks to predict protein secondary structure. *Proteins*, 19(1) :55–72, 1994.

[163] I. Ruczinski, C. Kooperberg, R. Bonneau, and D. Baker. Distributions of beta sheets in proteins with application to structure prediction. *Proteins*, 48(1) :85–97, 2002.

[164] A. A. Salamov and V. V. Solovyev. Protein secondary structure prediction using local alignments. *J. Mol. Biol.*, 268(1) :31–36, 1997.

[165] A. Sali and T. L. Blundell. Comparative protein modelling by satisfaction of spatial restraints. *J. Mol. Biol*, 234(3) :779–815, 1993.

[166] C. Sander and R. Schneider. Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*, 9(1) :56–68, 1991.

[167] R.A. Sayle and E.J. Milner-White. Rasmol : biomolecular graphics for all. *Trends Biochem Sci*, 20(9) :374, 1995.

[168] S Schbath. An efficient statistic to detect over- and under-represented words in dna sequences. *J. Comput. Biol.*, 4(2) :189–192, 1997.

[169] S Schbath, B Prum, and E de Turckheim. Exceptional motifs in different markov chain models for a statistical analysis of dna sequences. *J. Comput. Biol.*, 2(3) :417–437, 1995.

[170] M Schiffer and A B Edmundson. Use of helical wheels to represent the structures of proteins and to identify segments with helical potential. *Biophys. J.*, 7(2) :121–135, 1967.

[171] S. C. Schmidler, J. S. Liu, and D. L. Brutlag. Bayesian segmentation of protein secondary structure. *J. Comput. Biol.*, 7(1-2) :233–248, 2000.

[172] S.C. Schmidler, J.S. Liu, and D.L. Brutlag. Bayesian protein structure prediction. *Case Studies in Bayesian Statistic*, 5 :363–378, 2001.

[173] J. Schuchhardt, G. Schneider, J. Reichelt, D. Schomburg, and P. Wrede. Local structural motifs of protein backbones are classified by self-organizing neural networks. *Protein Eng*, 9(10) :833–42, 1996.

[174] G.E. Schulz, C.D. Barry, J. Friedman, P.Y. Chou, G.D. Fasman, A.V. Finkelstein, V.I. Lim, O.B. Pititsyn, E.A. Kabat, T.T. Wu, M. Levitt, B. Robson, and K. Nagano. Comparison of predicted and experimentally determined secondary structure of adenyl kinase. *Nature*, 250(462) :140–2, 1974.

[175] G. Schwartz. Estimating the dimension of a model. *Annals of Statistics*, 6 :461–464, 1978.

[176] J P Segrest, H De Loof, J G Dohlman, C G Brouillette, and G M Anantharamaiah. Amphipathic helix motif : classes and properties. *Proteins*, 8(2) :103–117, 1990.

[177] P. R. Sibbald and P. Argos. Weighting aligned protein or nucleic acid sequences to correct for unequal representation. *J. Mol. Biol.*, 216(4) :813–818, 1990.

[178] K.T. Simons, R. Bonneau, I. Ruczinski, and D. Baker. Ab initio protein structure prediction of casp iii targets using rosetta. *Proteins*, pages 171–6, 1999.

[179] K.T. Simons, C. Kooperberg, E. Huang, and D. Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions. *J Mol Biol*, 268(1) :209–25, 1997.

[180] K.T. Simons, I. Ruczinski, C. Kooperberg, B.A. Fox, C. Bystroff, and Baker D. Improved recognition of native-like protein structures using a combination of sequence-dependent and sequence-independent features of proteins. *Proteins*, 34(1) :82–95, 1999.

[181] M. J. Sippl. Recognition of errors in three-dimensional structures of proteins. *Proteins*, 17(4) :355–362, 1993.

[182] H. Sklenar, C. Etchebest, and R. Lavery. Describing protein structure : a general algorithm yielding complete helicoidal parameters and a unique overall axis. *Proteins*, 6(1) :46–60, 1989.

[183] J. Skolnick, A. Kolinski, D. Kihara, M. Betancourt, P. Rotkiewicz, and M. Boniecki. Ab initio protein structure prediction via a combination of threading, lattice folding, clustering, and structure refinement. *Proteins*, Suppl 5 :149–156, 2001.

[184] R. Srinivasan and G. D. Rose. Linus : a hierarchic procedure to predict the fold of a protein. *Proteins*, 22(2) :81–99, 1995.

[185] RE. Steward and JM. Thornton. Prediction of strand pairing in antiparallel and parallel beta-sheets using information theory. *Proteins*, 48(2) :178–91, Aug 2002.

[186] C. M. Stultz, J. V. White, and T. F. Smith. Structural analysis based on state-space modeling. *Protein Sci.*, 2(3) :305–314, 1993.

[187] C. Tarnas and R. Hughey. Reduced space hidden markov model training. *Bioinformatics*, 14(5) :401–406, 1998.

[188] T. Taylor, M. Rivera, G. Wilson, and I. I. Vaisman. New method for protein secondary structure assignment based on a simple topological descriptor. *Proteins*, 60(3) :513–524, 2005.

[189] W. R. Taylor. The classification of amino acid conservation. *J. Theor Biol.*, 119(2) :205–218, 1986.

[190] J. D. Thompson, D. G. Higgins, and T. J. Gibson. Clustal w : improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, 22(22) :4673–4680, 1994.

[191] M. J. Thompson and R. A. Goldstein. Predicting protein secondary structure with probabilistic schemata of evolutionarily derived information. *Protein Sci.*, 6(9) :1963–1975, 1997.

[192] J. L. Thorne, N. Goldman, and D. T. Jones. Combining protein evolution and secondary structure. *Mol. Biol. Evol.*, 13(5) :666–673, 1996.

[193] J. Tsai, R. Bonneau, A. V. Morozov, B. Kuhlman, C. A. Rohl, and D. Baker. An improved protein decoy set for testing energy functions for protein structure prediction. *Proteins*, 53(1) :76–87, 2003.

[194] R. Unger, D. Harel, S. Wherland, and J.L. Sussman. A 3d building blocks approach to analyzing and predicting structure of proteins. *Proteins*, 5(4) :355–73, 1989.

[195] C. Venclovas. Comparative modeling in casp5 : progress is evident, but alignment errors remain a significant hindrance. *Proteins*, 53 Suppl 6 :380–388, 2003.

[196] M. Vingron and P.R. Sibbald. Weighting in sequence space : a comparison of methods in terms of generalized sequences. *Proc Natl Acad Sci U S A*, 90(19) :8777–81, 1993.

[197] G. Wang and R.L. Jr. Dunbrack. Pisces : a protein sequence culling server. *Bioinformatics*, 19(12) :1589–91, 2003.

[198] J.J. Ward, L.J. McGuffin, B.F. Buxton, and D.T. Jones. Secondary structure prediction with support vector machines. *Bioinformatics*, 19(13) :1650–5, 2003.

[199] M W West and M H Hecht. Binary patterning of polar and nonpolar amino acids in the sequences and structures of native proteins. *Protein Sci.*, 4(10) :2032–2039, 1995.

[200] J. V. White, C. M. Stultz, and T. F. Smith. Protein classification by stochastic modeling and optimal filtering of amino-acid sequences. *Math Biosci.*, 119(1) :35–75, 1994.

[201] K.J. Won, A. Prugel-Bennett, and A. Krogh. Training hmm structure with genetic algorithm for biological sequence analysis. *Bioinformatics*, 2004.

[202] MA. Wouters and PM. Curmi. An analysis of side chain interactions and pair correlations within antiparallel beta-sheets : the differences between backbone hydrogen-bonded and non-hydrogen-bonded residue pairs. *Proteins*, 22(2) :119–31, Jun 1995.

[203] C. H. Wu, L.-S. L. Yeh, H. Huang, L. Arminski, J. Castro-Alvear, Y. Chen, Z. Hu, P. Kourtesis, R. S. Ledley, B. E. Suzek, C. R. Vinayaka, J. Zhang, and W. C. Barker. The protein information resource. *Nucleic Acids Res.*, 31(1) :345–347, 2003.

[204] K-P. Wu, H-N. Lin, J-M. Chang, T-Y. Sung, and W-L. Hsu. Hyprosp : a hybrid protein secondary structure prediction algorithm–a knowledge-based approach. *Nucleic Acids Res.*, 32(17) :5059–5065, 2004.

[205] Y. Xia, E. S. Huang, M. Levitt, and R. Samudrala. Ab initio construction of protein tertiary structures using a hierarchical approach. *J. Mol. Biol.*, 300(1) :171–185, 2000.

[206] A.-S. Yang and L.-Y. Wang. Local structure prediction with local structure-based sequence profiles. *Bioinformatics*, 19(10) :1267–1274, 2003.

[207] SM. Zaremba and LM. Gregoret. Context-dependence of amino acid residue pairing in antiparallel beta-sheets. *J Mol Biol.*, 291(2) :463–79, Aug 1999.

[208] A. Zemla, C. Venclovas, K. Fidelis, and B. Rost. A modified definition of sov, a segment-based measure for protein secondary structure prediction assessment. *Proteins*, 34(2) :220–3, 1999.

[209] W.M. Zheng. Clustering of amino acids for protein secondary structure prediction. *J Bioinform Comput Biol*, 2(2) :333–42, 2004.

[210] H. Zhu and W. Braun. Sequence specificity, statistical potentials, and three-dimensional structure prediction with self-correcting distance geometry calculations of beta-sheet formation in proteins. *Protein Sci*, 8(2) :326–42, Feb 1999.

[211] K Zimmermann. When awaiting 'bio' champollion : dynamic programming regularization of the protein secondary structure predictions. *Protein Eng.*, 7(10) :1197–1202, 1994.

Annexes

Annexe 1 : Méthode Rosetta

Prédiction de fragments

Des profils sont générés avec PSI-BLAST [4] pour la séquence à prédire ainsi que pour les protéines d'un sous-ensemble non redondants de structures de la PDB. La séquence à prédire (séquence cible) est découpée en fragments chevauchant de 3 et 9 résidus. Pour chaque fragment, des candidats structuraux sont extraits des structures de la PDB en utilisant une mesure de distance qui repose sur la comparaison des profils et la comparaison des structures secondaires (réelles pour les protéines de la PDB, prédites pour la séquence cible).

25 candidats structuraux sont retenus pour les fragments de 9 résidus et 200 pour les fragments de 3 résidus.

Une méthode hybride Rosetta-Isites a également été mise en place, dans laquelle la collection de fragments I-sites est utilisée pour fournir les candidats structuraux [35].

Assemblage

Fonction de score

Différents score sont utilisés aux différents stades de la simulation.

Le théorème de Bayes permet d'écrire :

$$P(structure/sequence) = \frac{P(sequence/structure) \times P(structure)}{P(sequence)}.$$

Terme P(séquence) Dans l'optique de rechercher la meilleure conformation pour une séquence donnée, le terme $P(sequence)$ est une constante, il est donc négligé.

Terme P(structure) Le terme $P(structure)$ ne dépend pas de la séquence à modéliser. Il peut être approximé par un **terme de compacité** [179] :

$$P(structure) = \exp(-r^2)$$

r étant le rayon de giration de la protéine : l'écart quadratique moyen des atomes lourds de la protéine à leur centre de masse. C'est donc une mesure de la compacité de la conformation proposée. Ce terme a pour but de favoriser les structures compactes car il

a été constaté que les structures générées aléatoirement sont beaucoup moins compactes que des structures de protéines naturelles.

P(structure) peut aussi être formulé comme un **terme de densité** autour des Cβ [180] :

$$P(structure) = \prod_i \frac{f(n_i)}{f_r c(n_i)}$$

où n_i est le nombre de Cβ dans une sphère de 10 Å de rayon autour du Cβ du résidu i, $f(n_i)$ est la fréquence empirique de n_i dans les structure natives et $f_r c(n_i)$ la fréquence empirique dans les structures aléatoires.

Ces termes ne sont cependant pas suffisants pour obtenir des arrangements corrects entre éléments de structure secondaire. Une façon d'en tenir compte est d'utiliser un **terme d'arrangement de paires de structures secondaires**. Ce terme, détaillé dans [180], utilise une représentation vectorielle des segments de structures secondaires et tient compte de la séparation et de l'orientation relative entre deux segments de structures secondaires.

Un **terme spécifique dédié à l'appariement des brins** β provient d'une étude approfondie des arrangements de brins β observés dans la PDB [163]. Cette étude montre notamment qu'un certain nombre d'arrangements possibles ne sont jamais utilisés.

Terme P(séquence/structure) Ce terme est approximé par [180] :

$$\begin{aligned} P(sequence \mid structure) &= P(aa_1 aa_2 aa_3 ... aa_n \mid structure) \\ &\approx \prod_i P(aa_i \mid struct) \prod_{i<j} \frac{P(aa_i, aa_j \mid struct)}{P(aai \mid struct) \times P(aa_j \mid struct)} \\ &\approx \prod_i P(aa_i \mid E_i) \prod_{i<j} \frac{P(aa_i, aa_j \mid r_{ij}, E_i, E_j)}{P(aai \mid r_{ij}, E_i, E_j) \times P(aa_j \mid r_{ij}, E_i, E_j)} \end{aligned}$$

La dépendance à la structure est exprimée en terme de séparation (distance r_{ij} entre deux résidus) et en terme d'environnement local E_i décrit par l'enfouissement du résidu.

Ces différents termes sont combinés en utilisant des pondérations optimisées pour discriminer les structures natives. Ils ne sont pas tous inclus dans la fonction de score à toutes les étapes de l'assemblage.

Simulation de Monte-Carlo

La procédure d'assemblage par Monte-Carlo peut se résumer en 5 étapes [30] :

1. La structure initiale est une conformation entièrement étendue. Les longueurs et angles de liaisons sont maintenus fixes au valeurs de référence observées dans l'alanine. Seuls les angles dièdres Φ et Ψ sont modifiables. La structure est modifiée par au moins 2000 insertions de fragments de 9 résidus, jusqu'à modification de tous les angles dièdres. La fonction de score utilisée à cette étape fait intervenir uniquement un **terme de gène stérique** pour empêcher les contacts trop proches entre atomes.
2. 2000 insertions de fragments de 9 résidus sont réalisées, en utilisant une fonction de score qui intègre le **terme P(sequence/structure)** et le **terme d'arrangement des structures secondaire**.
3. 10 itérations de 2000 insertions de fragments de 9 résidus sont réalisées pendant lesquelles le **terme d'appariement local de 2 brins** β est alternativement activé et désactivé. Ceci a pour but de favoriser l'appariement en brins β éloignés dans la séquence. **Le terme de densité** est également activé.
4. 3 itérations de 4000 insertions de fragments de 3 résidus sont réalisées, en ajoutant le terme de compacité et en utilisant un terme d'appariement des brins β plus élaboré.
5. Les filtres sont appliqués pour éliminés les mauvaises structures.

Le modèle simplifié de la protéine est enrichi au cours de la prédiction.

Choix du meilleur modèle

6000 à 150 000 modèles sont générés pour chaque séquence cible.

3 filtres sont utilisés pour éliminer les structures non natives :

1. Un filtre sert à éliminer les structures ayant un faible ordre de contact [23]. L'ordre de contact (CO) d'une structure est défini par

$$CO = \frac{1}{L \times N} \sum_{N} \Delta S_{ij},$$

L étant la longueur de la séquence et N le nombre de contacts entre résidus. Deux résidus i et j, séparés dans la séquence par $\Delta S_{ij} = |i - j|$ sont dits en contact si la distance entre leurs Cα est inférieure à 8 Å. Seuls les contacts tels que $\Delta S_{ij} \geq 5$ sont considérés.

2. Un filtre spécifique sert à éliminer les structures ayant des brins β mal appariés [163].
3. Après ajout des chaînes latérales, un filtre incluant les interactions de Van der Waals, la solvatation et les liaisons hydrogènes, est appliqué pour garder les meilleurs modèles [193].

Ces 3 filtres éliminent 30 à 90 % des modèles.

Les modèles restants sont classés par le RMSD (écart quadratique moyen des résidus alignés au mieux). Les séquences homologues de la cible sont également modélisées et classées. Le choix final du modèle repose sur l'hypothèse que le bassin d'énergie de la structure native est le plus large, et que c'est donc au voisinage de la structure native que l'on devrait trouver le plus de modèles générés. Les centres des plus grosses classes sont proposés comme modèle global de la prédiction.

Une expertise humaine, difficile à analyser était utile dans les premiers temps, pour choisir la structure finale [24]. La procédure Rosetta est aujourd'hui disponible sur serveur web de manière entièrement automatisée [102].

Annexe 2 : Loi stationnaire d'une chaîne de Markov

Définitions

Soit T_u est le temps de retour dans l'état u, défini par $T_u inf_{t\geq 1}\{t : S_t = u\} \mid S_0 = u$. L'état u est dit **récurrent** si et seulement si $\sum_{n=1}^{\infty} P(T_x = n) = 1$, autrement dit, il est certain que l'on reviendra dans l'état u après l'avoir quitté. Si $\sum_{n=1}^{\infty} P(T_x = n) < 1$, l'état est dit **transient**.

Si un état u est récurrent et qu'il conduit à v, alors v est récurrent et v conduit à u

Ces deux définitions permettent de partitionner l'espace des états en **classes de communication**, comme illustré par la figure 6.5.

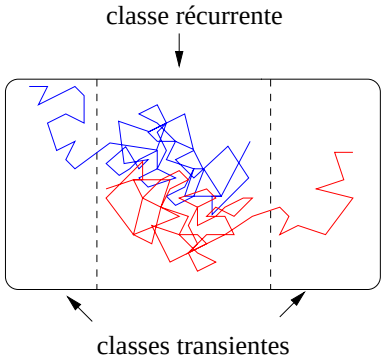

FIG. 6.5 – Représentation schématique de parcours dans un ensembe d'états partitionné en 3 classes de communication

Une chaîne de Markov est dite **irréductible** si l'espace d'états est constitué d'une seule classe de communication. Une chaîne de Markov est dite **récurrente irréductible** si l'espace des états est constitué d'une seule classe de récurrence.

Une chaîne de Markov est dite **périodique** si le chemin parcourt périodiquement des sous-ensembles de l'espace des états : les temps de retours dans les états sont périodiques.

Loi stationnaire

Soit une chaîne de Markov dont la matrice de transitions entre états cachés, de termes $a(u, v)$ est notée A. Si l'on note $\Pi(t)$ la loi des états cachés au temps t, d'éléments $\pi(u)$, alors la loi au temps $t + 1$ est donnée par la récurrence :

$$\Pi(t + 1) = \Pi(t)A$$

On démontre que si la chaîne de Markov est récurrente, irréductible, et apériodique, alors, $\Pi(t)$ converge vers une loi stationnaire unique Π indépendante de t.

Remarque : dans le cas d'un profil HMM, la matrice de transition entre états n'est pas une chaîne de Markov récurrente irréductible : le parcours dans le graphe des états est unidirectionnel, il n'y a pas de retour en arrière possible.

Annexe 3 : Complément pour le calcul de l'information portée par un arbre phylogénétique

Le calcul de l'information utilise $\Pi_a^g = P(g = a \mid \Phi)$, probabilité que le noeud a soit dans l'état a, sachant l'état de l'ensemble des feuilles de l'arbre.

$$\begin{aligned} \Pi_a^g &= P(g = a \mid \Phi) \\ \Pi_a^g &= \frac{P(g = a)P(\Phi \mid g = a)}{P(\Phi)} \text{ théorème de Bayes} \\ \Pi_a^g &= \frac{P(g = a)P(\Phi \mid g = a)}{\sum_b P(R = b)P(\Phi \mid R = b)} \end{aligned} \tag{6.1}$$

$$P(\Phi \mid g = a) = P(\Phi_g \mid g = a)P(\Phi - \Phi_g \mid g = a)$$

$\Phi - \Phi_g$ désignant l'état des feuilles de l'arbre n'étant pas portée par g. Le calcul de $P(\Phi - \Phi_g \mid g = a)$ nécessite de tenir compte des événements de mutations le long des branches entre g et les feuilles $\Phi - \Phi_g$.

Les notations nécessaires sont illustrées dans la figure 6.6. L'arbre est parcouru en remontant de g vers la racine : θ_g est le père de g, et θ_g^2, le père de θ_g. Le nombre d'intermédiaires entre g est la racine est désigné par $l(g)$. Le parcours est alors mené en redescendant depuis les noeuds θ : h_g désigne le fils de θ_g (hors lignée des θ).

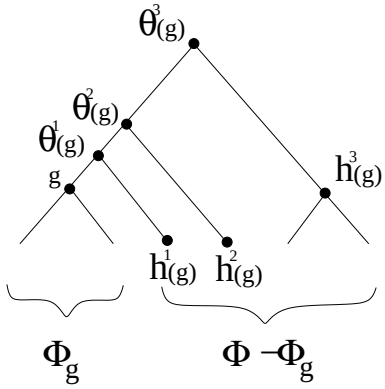

FIG. 6.6 – Représentation schématique de l'arbre phylogénétique

$$
\begin{aligned}
P(\Phi - \Phi_g \mid g = a) \quad = \quad & \sum_{\theta_g = a_1, \ldots, a_{l(g)} h_g = b_1 \ldots b_{l(g)}} \prod_{i=1}^{l(g)} P(\theta_g^i = a_i \mid \theta_g^{i-1} = a_{i-1}) \\
& \times P(h_g^i = b_i \mid \theta_g^i = a_i) \\
& \times P(\Phi_{h_g^i} \mid h_g^i = b^i)
\end{aligned}
$$

Les termes $P(\theta_g^i = a_i \mid \theta_g^{i-1} = a_{i-1})$ sont déduits des longueurs des branches, et $P(\Phi_{h_g^i} \mid h_g^i = b^i) = F_{b^i}^{h_g^i}$.

Annexe 4 : Articles, Posters et Communications

Articles

- Juliette Martin, Guillaume Letellier, Antoine Marin, Jean-François Taly, Alexandre G. de Brevern, Jean-François Gibrat. **Protein secondary structure assignment revisited : a detailed analysis of different assignment methods.** *BMC Structural Biology 2005 Sep 15 ;5(1) :17*
- Juliette Martin, Jean-François Gibrat, François Rodolphe. **How to choose the optimal hidden Markov model for protein secondary structure prediction.** *IEEE Intelligent Systems, Special issue on Data Mining for Bioinformatics, accepté, à paraître en novembre/décembre 2005*
- Juliette Martin, Jean-François Gibrat, François Rodolphe. **HMM for local protein structure.** *Proceedings of Applied Stochastic Models and Data Analysis, Brest 2005.* ISBN 2-908849-15-1. Editeurs : Jacques Janssen et Philippe Lenca.

Posters et présentations

- Juliette Martin, Pierre Nicolas, Jean-François Gibrat. **First step toward 3D protein structure prediction with a de novo approach.** $2^n d$ European Conference on Computational Biology (ECCB 2003) / Quatrièmes Journées Ouvertes de Biologie, Informatique et Mathématiques (JOBIM), Paris, 27-30 Septembre 2003. Poster
- Juliette Martin, Jean-François Gibrat, François Rodolphe. **Modèle de chaînes de Markov cachées pour la structure secondaire des protéines.** Cinquièmes Journées Ouvertes de Biologie, Informatique et Mathématiques (JOBIM 2004), Montréal, 28-30 Juin 2004. Poster
- Juliette Martin, Jean-François Gibrat, François Rodolphe. **Protein secondary structure prediction using hidden Markov models.** $12^t h$ International Conference on Intelligent Systems for Molecular Biology (ISMB 2004)/ $3^r d$ European Conference on Computational Biology (ECCB 2004), Glasgow, 31juillet-4 aout 2004. Poster
- Juliette Martin, Jean-François Gibrat, François Rodolphe. **Hidden Markov model for protein secondary structure prediction.** Integrative Post-Genomics (IPG'04), Lyon, 13-15 octobre 2005. Poster
- Juliette Martin, Jean-François Gibrat, François Rodolphe. **Prédiction de la struc-**

ture locale des protéines avec des modèles de chaînes de Markov cachées. Colloque du Groupe de Graphisme et de Modélisation Moléculaire (GGMM 2005) , Ile des Embiez, 2-4 mai 2005. Poster

- Juliette Martin, Jean-François Gibrat, François Rodolphe. **HMM for local protein structure.** Applied Stochastic Models and Data Analysis (ASMDA 2005), Brest, 17-20 mai 2005. Présentation orale accompagnant un article.
- Juliette Martin, Jean-François Taly, Jean-François Gibrat, François Rodolphe. **Choice of the optimal Hidden Markov Model for secondary structure prediction.** Sixièmes Journées Ouvertes de Biologie, Informatique et Mathématiques (JOBIM 2005), Lyon, 6-9 juilet 2005. Présentation courte accompagnant un poster.
- Jean-François Taly, Juliette Martin, Antoine Marin, Jean-François Gibrat. **Définition de mesures décrivant l'environnement local des acides aminés pour une application à l'évaluation des modèles structuraux.** Sixièmes Journées Ouvertes de Biologie, Informatique et Mathématiques (JOBIM 2005), Lyon, 6-9 juilet 2005. Présentation courte accompagnant un poster.

BMC Structural Biology

Research article

Open Access

Protein secondary structure assignment revisited: a detailed analysis of different assignment methods

Juliette Martin*[1], Guillaume Letellier[1], Antoine Marin[1], Jean-François Taly[1], Alexandre G de Brevern[2] and Jean-François Gibrat[1]

Address: [1]INRA, Unité Mathématiques Informatique et Génome, Domaine de Vilvert, 78352 Jouy en Josas Cedex, France and [2]INSERM U726, Equipe de Bioinformatique Génomique et Moléculaire, Université Paris 7, case 7113, 2 place Jussieu, 75251 Paris cedex 05, France

Email: Juliette Martin* - juliette.martin@jouy.inra.fr; Guillaume Letellier - guillaume.letellier@jouy.inra.fr; Antoine Marin - antoine.marin@jouy.inra.fr; Jean-François Taly - jean-francois.taly@jouy.inra.fr; Alexandre G de Brevern - debrevern@ebgm.jussieu.fr; Jean-François Gibrat - jean-francois.gibrat@jouy.inra.fr

* Corresponding author

Published: 15 September 2005

BMC Structural Biology 2005, **5**:17 doi:10.1186/1472-6807-5-17

Received: 26 May 2005
Accepted: 15 September 2005

This article is available from: http://www.biomedcentral.com/1472-6807/5/17

Abstract

Background: A number of methods are now available to perform automatic assignment of periodic secondary structures from atomic coordinates, based on different characteristics of the secondary structures. In general these methods exhibit a broad consensus as to the location of most helix and strand core segments in protein structures. However the termini of the segments are often ill-defined and it is difficult to decide unambiguously which residues at the edge of the segments have to be included. In addition, there is a "twilight zone" where secondary structure segments depart significantly from the idealized models of Pauling and Corey. For these segments, one has to decide whether the observed structural variations are merely distorsions or whether they constitute a break in the secondary structure.

Methods: To address these problems, we have developed a method for secondary structure assignment, called KAKSI. Assignments made by KAKSI are compared with assignments given by DSSP, STRIDE, XTLSSTR, PSEA and SECSTR, as well as secondary structures found in PDB files, on 4 datasets (X-ray structures with different resolution range, NMR structures).

Results: A detailed comparison of KAKSI assignments with those of STRIDE and PSEA reveals that KAKSI assigns slightly longer helices and strands than STRIDE in case of one-to-one correspondence between the segments. However, KAKSI tends also to favor the assignment of several short helices when STRIDE and PSEA assign longer, kinked, helices. Helices assigned by KAKSI have geometrical characteristics close to those described in the PDB. They are more linear than helices assigned by other methods. The same tendency to split long segments is observed for strands, although less systematically. We present a number of cases of secondary structure assignments that illustrate this behavior.

Conclusion: Our method provides valuable assignments which favor the regularity of secondary structure segments.

Background

In 1951, Pauling and Corey predicted the existence of two periodic motifs in protein structures: the α-helix [1] and the β-sheet [2] which turned out to be major features of protein architecture. Secondary structures, because they allow a simple and intuitive description of 3D structures, are widely employed in a number of structural biology applications. For instance, they are used for structure comparison [3] and structure classification [4,5]. They also provide a natural frame for structure visualization [6,7].

In recent years, secondary structures have come to play a major role in a number of methods aiming at predicting protein 3D-structures. Indeed, being able to predict accurately secondary structure elements along the sequence provides a good starting point toward elucidating the 3D-structure [8,9]. Current algorithms for predicting the secondary structure provides accuracy rates of about 80% for a 3 state prediction: α-helix, β-strand and coils [10-12], using neural networks and evolutionary information. The maximum achievable prediction has been estimated to lie in the range 85% [13] to 88% [14].

The divergence between observed and predicted secondary structure has been noticed early [15]. It took more time, though, for the structuralist community, to realize that obtaining an accurate and objective secondary structure assignment was not a trivial task, due to the variations observed in secondary structures when compared to ideal ones. As noted by Robson and Garnier [16]: "In looking at a model of a protein, it is often easy to recognize helix and to a lesser extent sheet strands, but it is not easy to say whether the residues at the ends of these features be included in them or not. In addition there are many distorsions within such structures, so that it is difficult to assess whether this represents merely a distortion, or a break in the structure. In fact the problem is essentially that helices and sheets in globular proteins lack the regularity and clear definition found in the Pauling and Corey models." For instance, as found by Barlow and Thornton [17] and Kumar and Bansal [18,19], a majority of α-helices in globular proteins are smoothly curved. Therefore, a group of experts (NMR spectroscopists and crystallographers), asked to assign the secondary structure of a particular protein, is likely to come up with different assignments.

To cope with this problem, as well as the increase in the number of experimentally solved 3D structures, the need for automatic secondary structure assignment programs was felt in the mid seventies. Such programs are intended to embody expert's knowledge and to provide consistent and reproducible secondary structure assignments. Periodic secondary structures generate regularities that can be used as criteria to define them, e.g., Cα distances, dihedral angles, like α angles or pairs of (Φ/Ψ) angles, and specific patterns of hydrogen bonds. Along the years, various methods using these criteria have been proposed. The first implementation of such methods, allowing automatic secondary structure assignment from 3D coordinates, was done by Levitt and Greer [20]. The algorithm was mainly based on inter-Cα; torsion angles.

A few years later, Kabsch and Sander developed a method called DSSP [21] that still remains one of the most widely-used program for secondary structure assignment. The DSSP algorithm is based on the detection of hydrogen-bonds defined by an electrostatic criterion. Secondary structure elements are then assigned according to characteristic hydrogen-bond patterns. This methodology has been widely accepted as the gold standard for secondary structure assignment. A number of software packages make use of DSSP when they need to assign secondary structures. For instance rasmol [6], the most widely distributed visualization software, assigns the repetitive structures with a fast DSSP-like algorithm. Similarly GROMACS analysis tools use the DSSP software [22].

STRIDE [23] is a software related to DSSP. It makes a very similar use of hydrogen-bond patterns to what is done in DSSP, although the definition of hydrogen-bonds is slightly different. In addition STRIDE takes into account (Φ/Ψ) angles to assign secondary structures. STRIDE is used by the visualization tool VMD [7] to assign secondary structures.

SECSTR [24] belongs to the same family of methods. It has been developed specifically to improve the detection of π-helices. Indeed, SECSTR's authors found dssp and STRIDE unable to detect several π-helices they were able to characterize with their method.

Other methods have been developed that use different criteria to assign secondary structures. DEFINE [25] relies on Cα coordinates only and compares Cα distances with distances in idealized secondary structure segments. It also provides a description of super-secondary structures. P-CURVE approach [26] is based on the definition of helicoidal parameters for peptide units and generates a global peptide axis. PSEA [27] only considers Cα atoms. It is based on distance and angle criteria. XTLSSTR [28] has been developed to assign secondary structures " in the same way a person assigns structure visually", from distances and angles calculated from the backbone geometry. It is concerned with amide-amide interactions. The most recent method, to the best of our knowledge, is VoTAP [29] which employs the concept of Voronoi tessellation, yielding new contact matrices.

Let us notice that structure files provided by the Protein Data Bank (PDB) [30] contain secondary structure descriptions in the HELIX, SHEET and TURN fields (see the PDB Format Description Version 2.2 [31]). These secondary structure descriptions are either provided by the depositor (optional) or generated by DSSP. Approximately 90% of the PDB files do have secondary structure fields. However, even though these fields are used, it may happen that only a few secondary structure elements, of interest for the depositor, are described, the others being ignored.

The variety of available methods illustrates the fact that there are several legitimate ways to define secondary structures. It is hardly surprising that these different methods provide different assignments, especially at the edges of secondary structure segments. For example, Colloc'h and co-workers [32] showed that the percentage of agreement is only 63% between DSSP, P-CURVE and DEFINE and that DEFINE tends to assign too many repetitive secondary structure segments. XTLSSTR authors noted that DSSP assigns more β-strands than XTLSSTR does [28]. SECSTR is logically more sensitive for π-helix detection than DSSP or stride [24].

In this paper we want to focus on how well some of the above methods handle the secondary structure irregularities mentioned by Robson and Gamier [16]. We are particularly interested in the way these different methods process the edges of secondary structure elements and deal with the various structure distorsions occurring in proteins. For structures solved by X-ray diffraction, it is well known that the resolution has a direct effect upon the quality of the resulting model. One expects the secondary structure assignment to be less accurate for low resolution structures [23]. It is thus interesting to assess the effect of the resolution upon the secondary structure assignment proposed by the different methods. It is also worth comparing secondary structure assignments for structures solved by X-ray crystallography and by NMR techniques. Structures solved by NMR correspond to proteins in solution and provide a more "dynamic" representation of the protein conformation than X-ray structures do. NMR structures are therefore more prone to local distorsions and constitute difficult, and interesting, cases for secondary structure assignment methods.

In the following we present a new method for secondary structure assignment, called KAKSI (KAKSI means "two" in Finnish) based on Cα distances and (Φ/Ψ) angles. These characteristics are intuitively used when examining visually a 3D structure. Our main purpose in developing this method was to deal, in a satisfactory way, with the structure irregularities. For instance we consider that regions of the polypeptide chain that show an abrupt change in their curvatures (such as kinks in *a* helices) should be considered as breaks in periodic secondary structures. The objective of an assignment method is to provide accurate and reliable assignment. Demonstrating that our methodology is an improvement over existing methods would be difficult since there is no standard of truth to benchmark methods with. We then carry out comparisons of the assignments of this new method with a number of other methods that use different criteria to define secondary structures: DSSP, STRIDE, SECSTR, XTLSSTR and PSEA, as well as with the descriptions found in PDB files. These comparisons are performed on 4 different datasets: 3 X-ray datasets with, respectively, high, medium and low resolution and an NMR dataset. This allows us to evaluate the effect of the resolution and experimental method upon the different secondary structure assignment methods.

We address the problem of inclusion of residues at the edges of helices and strands by examining the length of segments assigned by different methods. We also study the problem of correctly defining segments in case of distortions. More specifically, for helices, we appraise the geometry of helical segments using HELANAL [33], a software dedicated to this task.

Finally, we illustrate how KAKSI deals with distorted secondary structures by comparing its assignments with STRIDE assignments for a number of difficult cases.

Results and discussion

KAKSI parameters

In KAKSI secondary structure detection depends on a number of parameters (see Method section).

To test the robustness of the method to the choice of these parameters, we examined the effect of changing ε_H, ε_b and σ_b upon the secondary structure contents of the *comparison sets*. We let ε_H and ε_b vary in the range 1.29 to 3.30, and σ_b in the range 3 to 6. Each parameter is tested separately, while keeping other parameters to the selected values given in Methods section.

The effects are similar on all sets of structures. The decrease of ε_H below 1.96 results in a moderate diminution of the percentage of α-helix, whereas this percentage slightly increases when ε_H is greater than 1.96. Fewer β-sheets are assigned when ε_b, is lower than 2.58. On the contrary, the percentage of β-sheets increases when ε_b, is greater than 2.58. Slightly more β-sheets are assigned when σ_b is lower than 5, and there is a diminution of β-sheets assignment when σ_b is greater than 5.

Two different behaviors are observed: KAKSI assignments are not very sensitive to variations of α-helix detection

thresholds, but quite sensitive to variations of β-sheets detection thresholds. This is easily explained by the detection heuristic: the detection of α-helix is achieved by the distance or the angle criteria, moderate changes of ε_H are balanced by other criteria. On the contrary, the β-sheet detection is achieved by the satisfaction of both, distance and angle, criteria.

The two criteria implemented in KAKSI for kink detection in α-helices, K1 based on (Φ/Ψ) angles and K2 based on axes, are also tested. To evaluate the efficiency of each criterion, we analyze the geometry of kinked helices with the HELANAL software. We monitor the fraction of helices classified as kinked by HELANAL. This fraction is reduced when each criterion is used separately showing that both criteria are able to detect kinks (data not shown). Results obtained with K1 agree better with HELANAL results than those obtained with K2. However the best agreement with HELANAL is obtained when criterion K1 and K2 are used sequentially. Hereafter, KAKSI assignments are obtained with the parameter values given in Material and Methods and both criteria K1 and K2 applied for kink detection.

Secondary structure content

The secondary structure content is used to assess the sensitivity of different assignment methods to the structure resolution. Table 2 shows the secondary structure content in all our *comparison sets*, according to five available assignment softwares, KAKSI and the PDB description.

Table 2: Secondary structure content according to different assignment methods. %*H*: percentage of residues assigned in α-helix. %*b*: percentage of residues assigned in β-strand. See the text for β-strand assignment with *kaksi* using different parameter values on the *LRes* and the *NMR* sets.

Dataset	*HRes set*		*MRes set*		*LRes set*		*NMR set*	
Method	%H	%b	%H	%b	%H	%b	%H	%b
KAKSI	36.8	22.0	38.0	22.5	35.1	19.0	33.5	15.2
PDB	40.5	20.3	41.7	20.9	39.3	18.2	35.5	17.3
DSSP	35.9	22.5	37.3	22.9	35.4	20.4	32.2	17.3
STRIDE	36.4	22.6	38.6	23.3	36.3	21.2	33.7	18.8
PSEA	32.1	23.7	34.2	25.0	33.0	24.4	30.6	22.8
SECSTR	37.2	20.1	38.5	20.4	37.0	18.6	33.3	16.3
XTLSSTR	40.4	19.7	40.9	19.6	35.9	14.4	34.3	14.8

There is no absolute consensus, even for the *HRes set*, about secondary structure content according to different methods. STRIDE and DSSP figures are very close, as expected due to the similarity of these methods [21,23]. PSEA systematically assigns less helices and more strands than other methods. PDB assignments are always richer in α-helix than any automatic procedure. KAKSI assigns a fraction of periodic secondary structures comparable to STRIDE and DSSP on the *HRes set*.

Secondary structure contents in the *HRes* and the *MRes sets* are similar according to different methods. Assignments on the *LRes* and the *NMR sets* result in smaller contents in regular secondary structures. This is true for every assignment methods, but more or less marked, depending on the method. β-assignment is lower on the *LRes set* for a majority of methods. Only PSEA assignments show a proportion of β comparable for all datasets. It must be noted that this method consistently assigns more β-strands than all other methods, whatever the dataset considered. Overall, though, the influence of the resolution upon the assignments of the methods is moderate. The type of technique use to solve the structure (X-ray vs NMR) appear to have a more pronounced effect.

The decrease in β-sheets assignment on the *LRes* and *NMR sets* indicates that less stringent parameter values are required when dealing with structures belonging to these sets. For example, KAKSI assignment on the *LRes set* with $\sigma_b = 3$ result in a proportion of 22.3% residues in β-sheet and 20.7% with $\sigma_b = 3.30$ (data not shown). In the same way, the percentage of β-sheet residues in the *NMR sets* is about 17.7% with $\sigma_b = 3$ or $\varepsilon_b = 3.30$. Consequently, we suggest to adapt the β-sheet detection parameters when dealing with low resolution and NMR structures.

Measures of global agreement between methods

C_3 *scores*

Table 3 shows the C_3 scores obtained for the *HRes set* (the overall agreement between the different assignment methods show the same tendencies for the different *comparison sets*, [see Additional file 1]). A group of methods shows a strong agreement: C_3 scores within the group DSSP, STRIDE, SECSTR and PDB are all in the range 87.4% (SECSTR versus PDB) to 95.4% (STRIDE versus DSSP).

Table 3: C_3 scores between different methods on the HRes set

	DSSP	STRIDE	PSEA	SECSTR	XTLSSTR	PDB
KAKSI	82.1%	83.5%	81.5%	81.7%	78.3%	83.4%
DSSP		95.4%	80.1%	93.4%	80.4%	90.8%
STRIDE			81.1%	91.9%	80.8%	89.9%
PSEA				79.8%	75.8%	78.1%
SECSTR					79.6%	87.4%
XTLSSTR						80.7%

The strong similarity between DSSP and STRIDE assignments, which both used a hydrogen-bond criterion, has been noted in previous studies [27,29,34]. The SECSTR method is strongly related to the DSSP algorithm and logically belongs to this group. As was expected, PDB descriptions are very close to DSSP assignments due to the way secondary structure assignments are performed.

Assignments given by XTLSSTR are the most different from others: C_3 scores with DSSP, STRIDE, SECSTR and PDB are all below 81%. KAKSI and PSEA show an intermediate behavior of the other methods [see Additional file 2]. The C_3 scores are all in the same range, between 81.5% (KAKSI/PSEA) and 83.5% (KAKSI/STRIDE), excluding XTLSSTR (78.3%).

SOV criterion

The SOV criterion is usually employed for secondary structure prediction evaluation, whereas here, comparisons are made between alternative structure assignments. SOV values depend on which structure ischosen as reference. To allow comparison, KAKSI is taken as reference. Table 4 shows SOV values computed from the *HRes set* for helices and strands, between KAKSI and other methods. SOV values for other datasets are available, [see Additional file 3].

Table 4: SOV measures between kaksi and other methods on the HRes set. SOV_H: SOV for α-helix. SOV_b: SOV for β-strand. KAKSI is taken as reference.

Method	SOV_H	SOV_b
DSSP	91.7%	92.1%
STRIDE	91.2%	91.9 %
SECSTR	89.0%	83.9%
PSEA	87.5%	82.7%
XTLSSTR	89.3%	73.4%
PDB	88.4%	89.4%

For helical segments, the highest SOV with KAKSI assignment is obtained with DSSP (91.7%). It lies in the same range for STRIDE. It is slightly lower for other methods but remains above 87%. For the strands, a good agreement is seen with DSSP, STRIDE and PDB (SOV scores about 90%). Lower SOV (about 83%) are found with PSEA and SECSTRC. Moderate agreement is seen with XTLSSTR (75.8% only). C_3 score between XTLSSTR and KAKSI is only 78.3%(see table 3). SOV values are high for helices and slightly lower for strands, showing that differences between both methods mainly concern β-sheets assignments. Hereafter we will restrict our comparisons to KAKSI, STRIDE, and PSEA assignments on the *HRes set*. STRIDE is a widely-used method whose results are very similar to DSSP and PDB, as shown by the C_3 scores. STRIDE is chosen because it exhibits the largest C_3 score with KAKSI. PSEA is chosen because its algorithm fairly differs from other methods, but SOV values remain consistent when compared to KAKSI'S.

Segment length distribution

The length distributions of helices and strands assigned by KAKSI, PSEA and STRIDE on the *HRes set* are shown on Figure 3.

In helix distributions, three zones can be distinguished. (i) For helices shorter than 8 residues, the distributions are very different: STRIDE assigns many 3 residue long

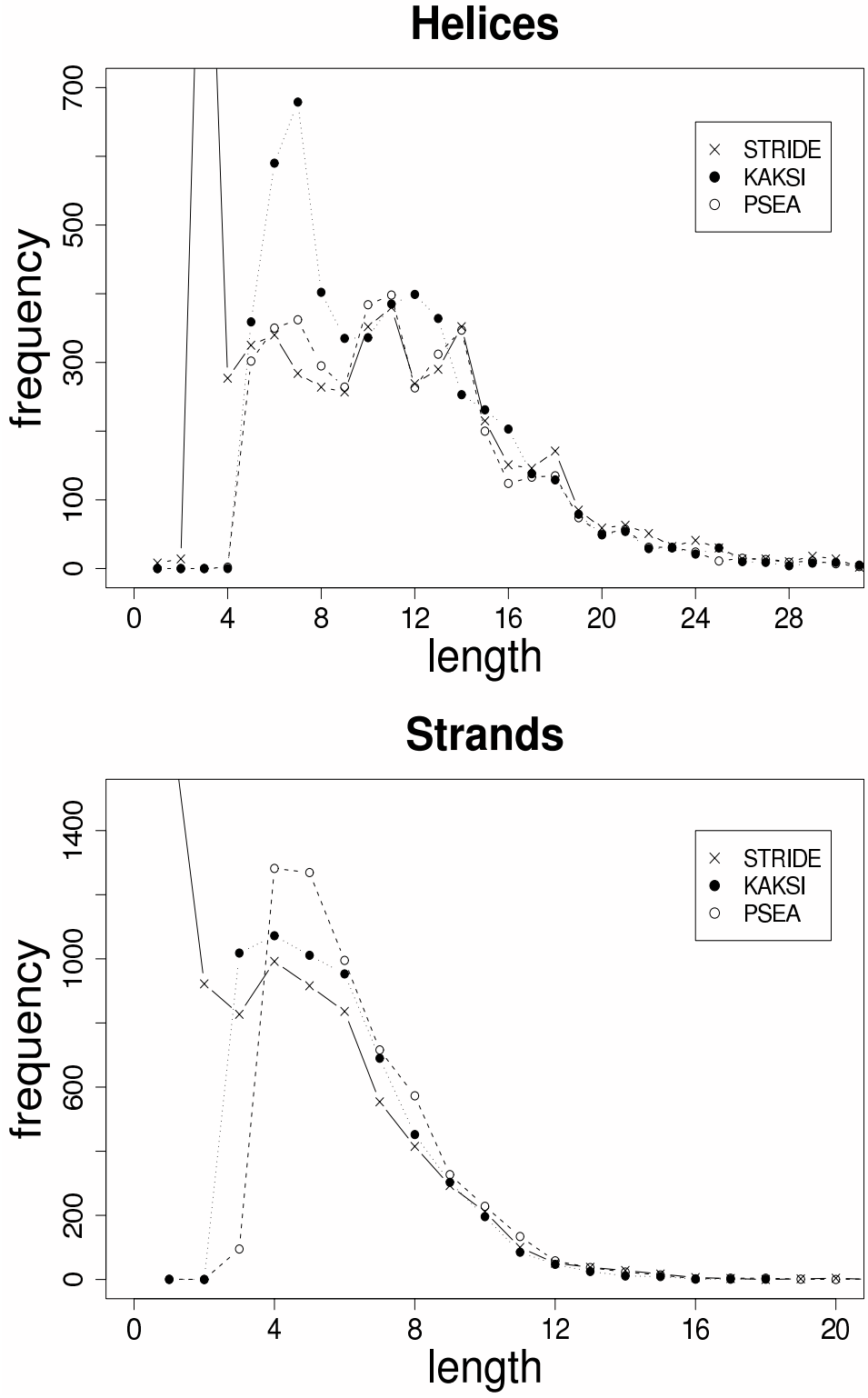

Figure 3
Length distribution of helices and strands assigned by stride, psea and kaksi. Length distribution of helical (top) and extended (bottom) segments assigned by STRIDE (plain line and crosses), PSEA (dashed line and open circles), and KAKSI (dotted line and filled circles), on the *HRes set*. The STRIDE assignment generates a large number of 3 residue-long helices (1238 segments) and 1 residue-long strands (corresponding to 1800 β-bridges).

helices, whereas PSEA and KAKSI do not assign helices shorter than 5 residues. PSEA assignments results in slightly larger number of short helices than STRIDE. KAKSI distribution shows a very high peak at 7 residues. (ii) In the range 8 to 15 residues, small differences are observed: KAKSI distribution shows a peak about 12 residues, unlike PSEA and STRIDE distributions. (iii) For helices longer than 15 residues, distributions are similar.

Similarly, 3 distinct zones appear in the strand distributions. (i) Up to 6 residues, PSEA and KAKSI curves show larger peaks than STRIDE distribution, at 3 to 5 residues for KAKSI, and 4 and 5 residues for PSEA. PSEA and KAKSI do not assign strands shorter than three residues, whereas STRIDE assignment result in a large number of 1-residue long strands. These segments are isolated β-bridges (state b in stride assignments). (ii) Between 6 and 9 residues, psea and KAKSI segments are more numerous than STRIDE segments. (iii) After 9 residues, the distributions are identical.

Global measures, such as C_3 and SOV scores, show that KAKSI assignments are globally consistent with those given by other existing methods. The length distributions of helices and strands indicates that segment distribution is also roughly similar across methods. This broad consensus was expected. In the following sections we now turn toward the study of details of the assignments, in particular, as mentioned in the introduction, we compare the way different methods deal with the edges of secondary structures and cope with local distorsions.

Detailed comparison

Pair length

The SOV criterion is a measure of the global overlapping of secondary structure segments. It gives no information about the effect of length of segments or about the respective length of facing segments. Figure 4 shows the plot of lengths for pair of corresponding repetitive structure segments between STRIDE and KAKSI, and PSEA and KAKSI assignments. The pairs are those used for the SOV computation: a pair is considered when there is at least one residue in the same state for the two assignments. Unpaired segments are ignored.

Taking KAKSI assignment as our reference, three different cases occur: (i) One segment according to KAKSI corresponds to a single segment in another method assignment: these are *one-to-one events*. (ii) One segment assigned by KAKSI corresponds to two or more segments in another method assignment. We call this a *fusion event*. (iii) The symmetric case, several segments in KAKSI assignment corresponding to a single segments in another method assignment, is called a *division event*. The three cases are available plotted on separate graphs [see Additional file 4].

Helix length

The strong accumulation of points along the diagonal, on both plots (KAKSI versus STRIDE and KAKSI versus PSEA) and for every segment lengths shows that KAKSI often agrees with other methods about the length of helices. There are more points below the diagonal than above, indicating that KAKSI tends to assign slightly longer segments than STRIDE and PSEA (one or two residue longer). This occurs for all segment lengths, but it is more striking on the PSEA/KAKSI comparison.

The points appearing far from the diagonal correspond to *division* and *fusion events*, as shown by the squared correlation coefficients r^2. Correlations are calculated on the pairs (PSEA or STRIDE length/KAKSI length) and are used as indicators for the dispersion about the diagonal. On the KAKSI/STRIDE comparison, $r^2 = 0.28$ for all the 5146 pairs, but reaches 0.88 when only the 3755 *one-to-one events* are considered. The remaining pairs correspond to 142 cases of *fusion* and 1249 cases of *division events. Division events* are responsible for the numerous observations of pairs of short helices in KAKSI assignment (5 to 9 residues) with longer helices in PSEA and STRIDE assignments (10 to 20 residues).

Similarly, for the KAKSI/PSEA comparison there are 4762 pairs ($r^2 = 0.23$), distributed in 3443 *one-to-one events* ($r^2 = 0.85$), 150 *fusion* and 1169 *division events*. Numerous cases of divisions appear on the plot as pairs of 5 to 9 residue helices for KAKSI and 10 to 20 residue helices for PSEA.

For both comparisons (KAKSI/STRIDE and PSEA/KAKSI), the number of *division events* is greater than the number of *fusion events*, showing that KAKSI tends to split long segments into shorter ones. This is a direct consequence of the kink detection mechanism used in KAKSI. It also explains why short helices are more abundant in KAKSI assignments than in STRIDE and PSEA. Some examples of this phenomenon are illustrated in Fig 5.

Strand length

The situation is less clear than for helices. The points are more dispersed and there is no clear accumulation of points accounting for *division events*. In the KAKSI/STRIDE comparison, the 5974 pairs yield a r^2 equal to 0.35. This value increases to 0.69 when only the 5403 *one-to-one events* are considered. Amongst the remaining pairs 214 correspond to *fusion events*, and 357 to *division events*. The splitting of long segments is thus less systematic than for helices. This makes senses since there is no mechanism similar to the kink detection in helices for β-strands. 52% of the *one-to-one events* fall above the diagonal (longer

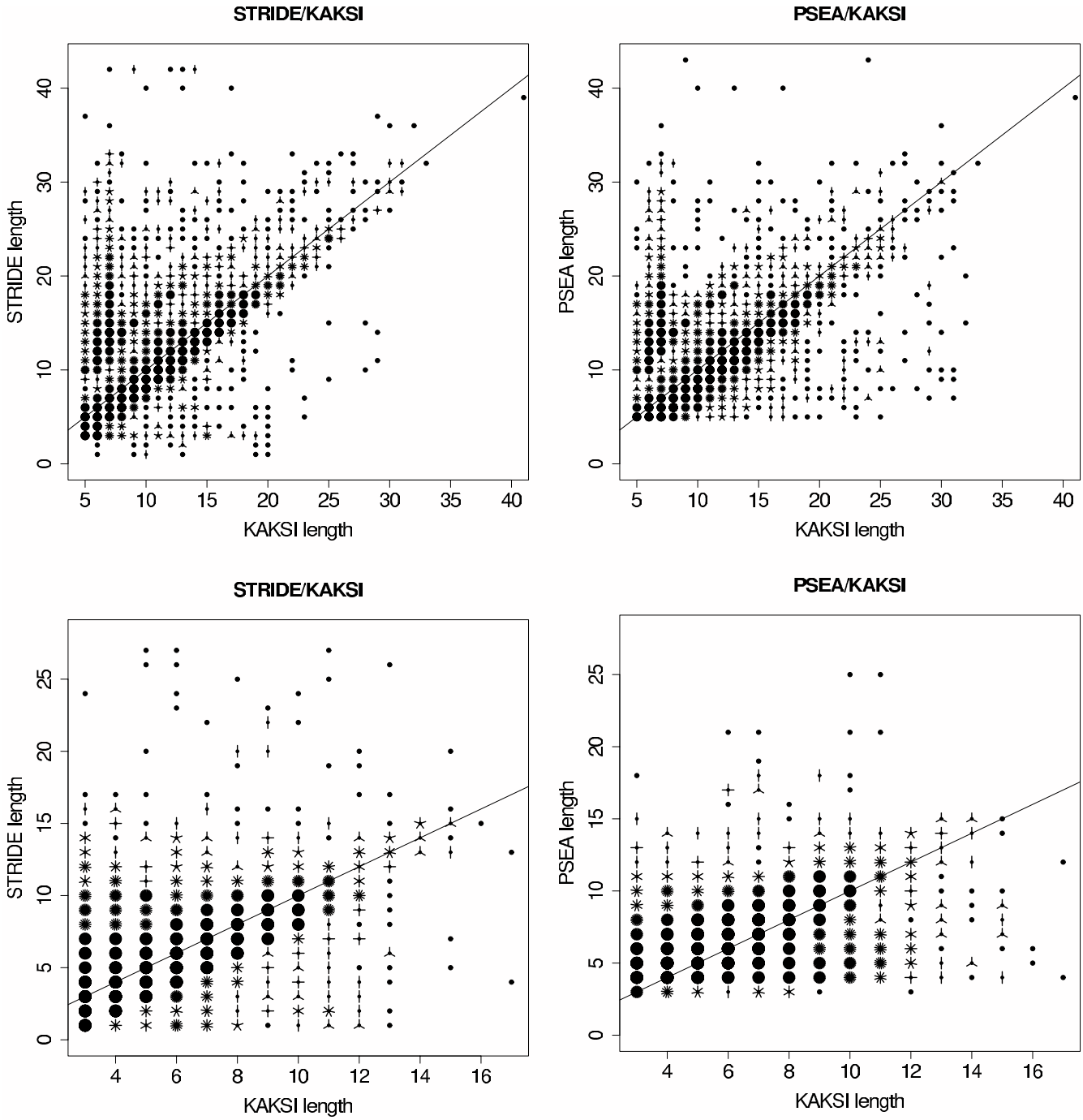

Figure 4

Length for pair of segments assigned by stride vs kaksi and psea vs kaksi. Length for pair of helices (upper part) and strands (lower part) when comparing STRIDE and KAKSI assignments, and PSEA and KAKSI assignments. We report a pair when we found at least one residue in the same state in both assignments. Data are shown as a "sunflower plot": a point stands for a single observation, then the number of "leaves" is proportional to the number of additional observations. The diagonal x = y (same length for two assignments) is shown.

segments in KAKSI assignment) and 22 % fall below the diagonal (shorter segments in KAKSI assignment). The remaining 26% are on the diagonal. It shows that KAKSI tend to assign longer strands than STRIDE.

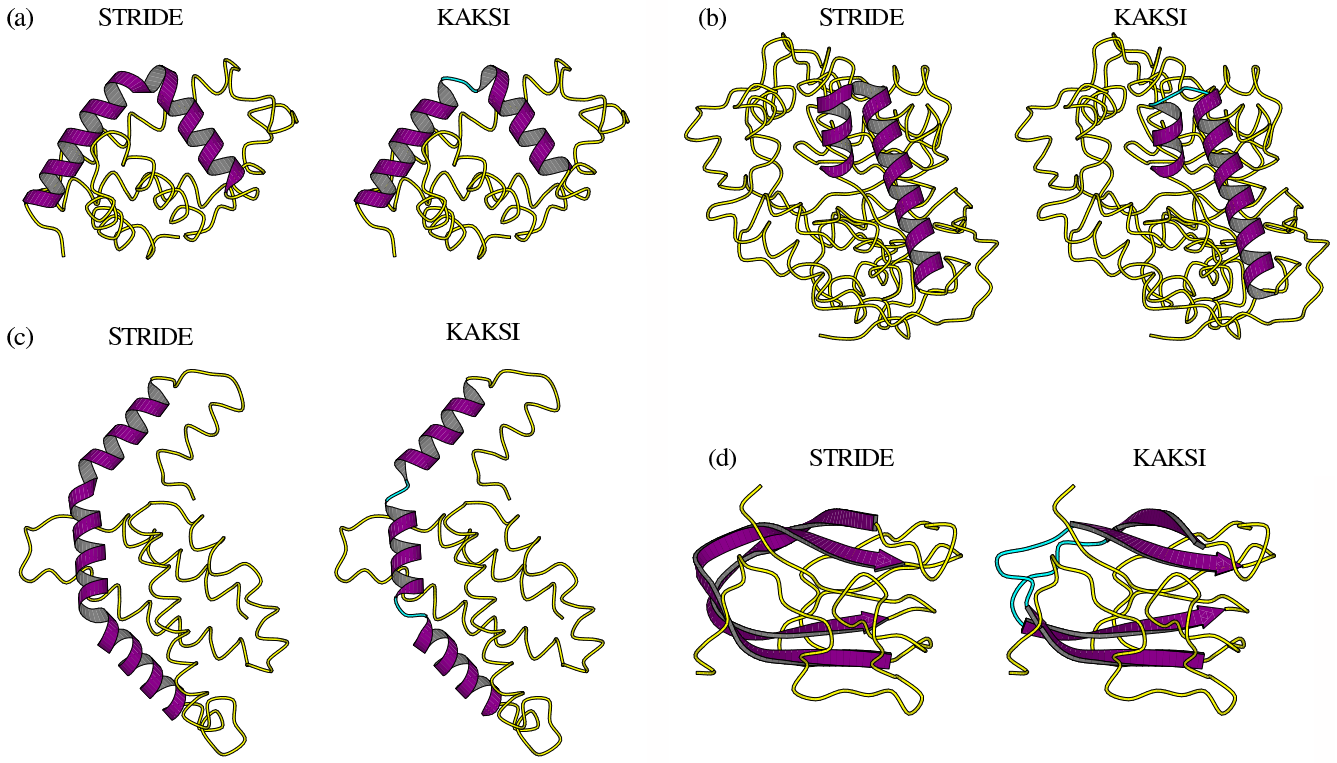

Figure 5

Examples of disagreement between kaksi and stride. The divergent assignments are drawn in cartoon representation and highlighted in purple (helix and strand) and cyan (coil assigned by KAKSI). Images are generated with Molscript [46]. Average bending angles (*AverBA*) between local axes computed by HELANAL in long helices are reported, (a): hemoglobin I from the clam Lucina pectinata, PDB code:1b0b, resolution 1.43 Å. STRIDE assignment: α-helix from residues 4 to 35, *AverBA* = 15.4°. KAKSI assignment: two helices from 4 to 19, *AverBA* = 3.84° and 21 to 34, *AverBA* = 9.0°. (b): chain A of L(+)-mandelate dehydrogenase from Pseudomonas *putida*, PDB code: 1p4c, resolution 1.35 Å. STRIDE assignment: helix from 308 to 340, *AverBA* = 24.7°. KAKSI assignment: two helices from 308 to 315 and 320 to 341, *AverBA* = 4.3°. (c): chain B of C-phycocyanin from the thermophylic cyanobacterium Synechococcus *elongatus*, PDB code: 1jbo, resolution: 1.45 Å. STRIDE assignment: helix from residues 21 to 62, *AverBA* = 13.1°. KAKSI assignment: 3 helices from 21 to 33, *AverBA* = 4.5°, 35 to 46, *AverBA* = 3.0°, and 48 to 61, *AverBA* = 6.6°. (d): chain A from endo-xylanase from Clostridium *stercorarium*, PDB code: 1od3, resolution: 1 Å. STRIDE assignment: two β-strands from 61 to 82 and 116 to 135. KAKSI assignment: four β-strands from 61 to 69, 75 to 83, 115 to 122, and 128 to 136.

In the KAKSI/PSEA comparison, r^2 equals 0.23 on the 5041 pairs and 0.44 on the 4694 *one-to-one events*. There are 214 *fusion events* and 133 *division events*. The numbers of *division* and *fusion* events are close, indicating that there only a slight splitting effect. 27% of the *one-to-one events* are on the diagonal, 50% are above (greater length in PSEA assignment) and 23% are below (greater lenght in *kaksi* assignment). In a majority of case, KAKSI assigns shorter strand segments concerning *one-to-one events*.

For both kind of segments and both comparisons, we also checked for the existence of systematic shifts of the segments toward the N-ter or C-ter termini of the secondary structure elements. No such systematic bias was found (data not shown).

Helix geometry analysis with HELANAL

In KAKSI we pay a special attention to the detection of kinks in α-helices by applying angle and axis criteria. This motivates the study of the geometry of helices with an external tool, according to alternative definitions of helix locations. We check the geometry of helices assigned by the different assignment methods with the HELANAL software. We are interested in the distribution of helices into the three classes: linear (L), curved (C) or kinked (K). Unclassified helices represent less than 1% in our datasets.

When analyzed by HELANAL, helices assigned by all methods show a high proportions of kinks. On the *HRes set*, for example, about 20% (DSSP, STRIDE, KAKSI) up to 30% (SECSTR, XTLSSTR) helices appear classified as kinked. This ratio is 16% only for the PDB assignments, and less than 10% for PSEA. When the resolution gets worse, this proportion increases [see Additional file 5]. On the *NMR set*, we observe as much as 40% kinked helices for PSEA assignment and more 50% kinked helices for STRIDE, SECSTR and PDB.

This high ratio of irregular helices (curved or kinked) is in agreement with previously published results [17]. However, the high ratio of kinked helices found here is larger than previously reported by Kumar and Bansal [19]. There is a difference between Kumar and Bansal's work and our study: they modified helix assignment given by DSSP before submission to HELANAL. Using distance and axis criteria, they corrected helix boundaries to avoid distortions at the termini. Consequently, the high ratio of kinked helices is likely due to these terminal residues. Rather than applying the correction used by Kumar and Bansal, we apply a systematic correction before submitting helices to HELANAL, i.e., one residue is removed at each helix terminus. The reason for applying a systematic correction rather than a correction based on geometrical criteria is that we want to make a statistical comparison of helices assigned by various softwares. The goal is not to correct potentially wrong helices boundaries. We want to evaluate the assignments as they are produced by the softwares and used in later applications.

Table 5 shows the results obtained on the *HRes set*, before and after correction, for helices defined by the seven methods. Results for other datasets are available [see Additional file 5].

Table 5: Helix geometry analyzed by HELANAL on the HRes set. Correction: assignments are corrected by shortening each helix by one residue at each terminus. %L: percentage of helices that are linear according to HELANAL. %C: percentage of helices that are curved according to HELANAL. %K: percentage of helices that are kinked according to HELANAL. N: number of helices submitted to HELANAL.

Method	No correction				With Correction			
Minimum length	11				9 after correction			
	%L	%C	%K	N	%L	%C	%K	N
DSSP	8.3	70.0	21.2	2215	10.9	70.8	17.8	2215
STRIDE	10.1	65.9	23.6	2431	10.8	68.5	20.2	2431
PSEA	10.9	78.5	10.0	2260	11.5	80.0	7.8	2260
SECSTR	8.0	55.7	36.0	2349	10.0	59.7	29.9	2349
XTLSSTR	8.7	58.9	32.1	2618	9.5	61.4	28.9	2618
KAKSI	10.2	66.5	22.8	2442	12.3	72.6	14.5	2442
PDB	11.4	71.1	17.0	2565	11.3	71.5	12.0	2565

As HELANAL can handle only helices longer than nine residues, we restrict our analysis to helices longer than eleven residues. When removing the first and last residues of helices, the ratio of kinked helices decreases, showing that part of the kinks are due to distortion at the termini. After correction, the geometry of helices assigned by KAKSI (14.5% of kinked helices) is the closest to the geometry of helices described in the PDB (12% kinked helices). The KAKSI method also assigns the highest ratio of linear helices (12.3%). PSEA has only 7.8% kinked helices but it should be noted that the number of helices submitted to analysis is slightly lower.

It is interesting to investigate the geometry of helices when KAKSI assigns several helices in a region where STRIDE assign a single long helix, i.e., the *division events*. If we consider the *division events* involving pair of helices longer than nine residues, we find 128 pairs where a kinked helix assigned by *stride* corresponds to curved or linear helices assigned by KAKSI. The symmetric case, kinked helices in KAKSI assignment paired with a curved or linear helices in STRIDE assignment concerns only 7 cases. This indicates that splitting long helices into several short ones helps to define helices devoid of kink.

All these observations suggest that the kink detection implemented in KAKSI is efficient and leads to more reliable helix locations. The major feature of KAKSI assignments is then the geometry of α-helices: while assigning slightly longer helices than stride, the global geometry of helices remains satisfactory, with more linear helices than other assignments and a limited ratio of kinked helices,

very close to PDB assignments. This is accomplished by dividing long distorted helices when appropriate. Some examples are shown in the following section.

Some examples of assignment disagreements

Figure 5 shows some interesting examples of disagreement between STRIDE and KAKSI assignments. The first three examples in Figure 5 concern disagreement about helix assignments. In example (a), the long helix assigned by STRIDE shows a sharp kink. In KAKSI assignment it is replaced by two helices from residues 4 to 19 and 21 to 34. The first helix is classified as curved by HELANAL. The second one is classified as kinked, but it becomes linear after removal of terminal residues. The angle between two global axes fitted in these two helices is 83°. The second example (b), is even more striking: a 33-residue long helix defined by STRIDE from residues 308 to 340 exhibits a reverse turn near its N-terminal edge. The definition given by KAKSI is two helices from 308 to 315 and 320 to 341. The first helix is too short to be analyzed by HELANAL and the second one is classified as linear. The third example is the case of a division of a long helix assigned by STRIDE into three segments in KAKSI assignment. Although less marked than for the first two examples, the kinks are well apparent. The three helices defined by KAKSI are all classified as curved by HELANAL, with their global axes making angles equal to 135 and 120° between the first and the second, and the second and the third helix respectively.

The last example 5(d) is an example of disagreement on a β-strands assignment. β-strands assigned by STRIDE are fairly curved, allowing a change of direction of the backbone. No specific routine is implemented in KAKSI to split distorted strands, as it is done for helices. Nonetheless, the criteria of β-sheet assignment being fairly strict, some cases of division in long β-strands can also occur. These examples illustrate the fact that a small disagreement on a per-residue basis can result in a radical change in the structure description. In the examples shown on Fig. 5 we believe that KAKSI assignments provide a more pertinent description of the protein structure.

Conclusion

We have developed a new automatic procedure to assign secondary structures from 3D coordinates. Our method, KAKSI, uses Cα distances and (Φ/Ψ) angles and pay a special attention to kink detection in helices. Like other methods (except PSEA), it is sensitive to the resolution, and the type of experimental technique used to solve the structure. Consequently, we propose to choose detection parameters according to the structure resolution or technique and the nature of the secondary structure, since β-sheets are more difficult to detect. The careful comparison of KAKSI assignments with assignments produced by five available methods and the description provided by the PDB highlights the similarities and differences between the different methods. Good general agreement are observed between methods, especially on α-helices. The length of α-helices and β-strands, in case of agreement on the number of segments, are very similar when compared to STRIDE and PSEA. When different lengths are assigned, we observe slightly longer α-helices and β-strands than the STRIDE definition. When two methods disagree on the number of segments, we observe more *division events* than *fusions*, i.e., several short helices assigned by KAKSI in front of a unique long helix assigned by STRIDE or PSEA. *Division events* are also slightly predominant in the comparison of β-strand length with STRIDE and PSEA. The study of α-helix geometry with an external tool reveals that KAKSI helices are less kinked that helices assigned by other methods, except PSEA. KAKSI is also the method that assigns helices with geometrical characteristics in best agreement with helices described in the PDB, and, maybe more important, the highest proportion of linear helices. As stated by Andersen and co-workers [35], each method reflects its own definition of secondary structures. Our definition favors a certain regularity of secondary structure elements, as illustrated by the examples on Fig. 5.

Methods

Datasets

The KAKSI method uses geometrical characteristics of α-helices and β-sheets extracted from available protein structures. A reference set (*Ref set*), consisting of 2880 structural domains taken from ASTRAL 1.63 [36] is used to estimate these geometrical characteristics. The list of domains with less than 40% identity provided by the ASTRAL server [37] is filtered to keep only X-ray structures with a resolution better than 2.25 Å and longer than 50 residues.

KAKSI assignments are compared with secondary structure assignments done by other methods. For the reasons mentioned above four different sets of structures are used. Hereafter we refer to these datasets as the *Comparison sets*.

The number of structures reported below refer to the files that are successfully processed by all assignment programs and contain a secondary structure description provided by the PDB.

• A High Resolution set (*HRes set*): X-ray structures with resolution better than 1.7 Å, R-factor < 0.19, identity percentage between sequences less than 30%, obtained from the WHATHIF website [38,39]. There are 689 structures in this set, corresponding to 151922 residues with a defined secondary structure, i.e., excluding missing coordinates.

• A Medium Resolution set (*MRes set*): X-ray structures with resolution between 1.7 Å and 3 Å, R-factor < 0.3, identity percentage between sequences less than 30%, minimum length of 40 residues, provided by the PISCES website [40,41]. There are 624 structures in this set, corresponding to 160 276 residues with a defined secondary structure.

• A Low Resolution set (*LRes set*): X-ray structures with resolution worse than 3 Å, R-factor > 0.3, identity percentage between sequences less than 30%, minimum length of 40 residues, provided by the PAPIA website [42]. There are 332 structures in this set, corresponding to 97852 residues with a defined secondary structure.

• A *NMR set*: structures with less than 30% sequence identity, extracted from all NMR entries obtained on the PDB website [43]. The redundancy of the set is reduced to 30% sequence identity with PISCES. There are 296 structures in this set, corresponding to 27533 residues with a defined secondary structure.

These lists are available on the web [see Additional file 6].

KAKSI method

The assignment of repetitive secondary structures by KAKSI is based on a set of characteristic values of Cα distances and (Φ/Ψ) dihedral angles. The parameters of KAKSI have been chosen to best fit the secondary structure assignments obtained from the PDB files (HELIX and SHEET fields). These fields, when present, are automatically generated with the DSSP method or are provided by the depositor who might have used some secondary structure assignment program and/or might have inspected visually the 3D structure and assigned himself the secondary structures. We use these PDB assignments as our gold-standard for the sake of parameter calculations, keeping in mind that the data are partly similar to DSSP assignments. Assignment is done by sliding windows along the sequence. α-helices are assigned first, followed by β-sheets. Two windows are slid for the β-sheet detection because we only want to assign β-strands involved in β-sheets. Residues once assigned in α-helix cannot be re-assigned in β-sheets.

Secondary structure characteristics used by the KAKSI heuristic

As mentioned earlier, α-helices and β-strands being periodic structures, their backbone geometry exhibits a number of regularities. This periodicity leads to characteristic distances between Cα atoms as well as characteristic values of (Φ/Ψ) dihedral angles.

More precisely, we have estimated from the *Ref set*:

• *distances between Cα in α-helices and β-sheets*. Different statistical distributions are computed for terminal residues and cores of secondary structure segments because greater variations are observed at segment termini. For α-helices, 4 distances are considered between residues i and j along the sequence, with $j \in [i + 2, i + 5]$. Table 1 shows the means and standard deviations obtained on the *Ref set*. For β-sheets, three different types of distances are considered. Figure 1 illustrates these distances and reports the values obtained on the *Ref set*.

• (Φ/Ψ) *values for residues involved in α-helices and β-strands*. Densities of (Φ/Ψ) angles are computed using Ramachandran maps. These maps are divided into 10 by 10 degree squares. This yields two *population maps*: one specific of α-helices and the other specific of β-strands [see Additional file 7]. For the α-helix *map*, we only consider angles lying in the area ($\Phi < 0°$ and $-90° < \Psi < 60°$) and we set to zero square frequencies that are too low (frequency $< \delta_H$). In this study, the threshold δ_H is fixed, empirically, to $20 \times n_{mean}$, n_{mean} being the mean frequency for a square in the Ramachandran map.

As mentioned above we are particularly interested in the detection of kinks in α-helices. Kinks are frequent and not easy to detect with usual distance and angle criteria. In a regular helix, (Φ/Ψ) angles should remain located in a narrow region of the Ramachandran map. One way to detect kinks (criterion K1 below), is to compute distances between (Φ/Ψ) pairs of successive residues j and $j + 1$ in the Ramachandran map. We use the 95-percentile of the distance distribution in α-helices. The kink detection is only performed in helix cores, terminal residues of segments being disregarded in the computation.

KAKSI heuristic for helix and strand assignment

Figure 2 illustrates the heuristic implemented in KAKSI. We have tested several criteria and combinations of criteria. The final heuristic presented here shows a good agreement with PDB assignments. The principle of the assignment is to test the Cα distances along the protein to check if they are close to the typical distances in regular secondary structure. The (Φ/Ψ) angles are tested in the same manner. α-helix assignment is achieved according to a distance or an angle criterion. The β-sheet detection requires the satisfaction of both angle and distance criteria. α-helix assignments are corrected whenever kinks are detected. Criteria applied at each step shown on Figure 2 are explained below, in the order they appear in the assignment process. Characteristic values extracted from the *Ref set* are shown in capital. The parameters of the method are : ε_H and ε_b are used to define thresholds for Cα distances and η_H and σ_b are used to define thresholds for the constraints on (Φ/Ψ) angles.

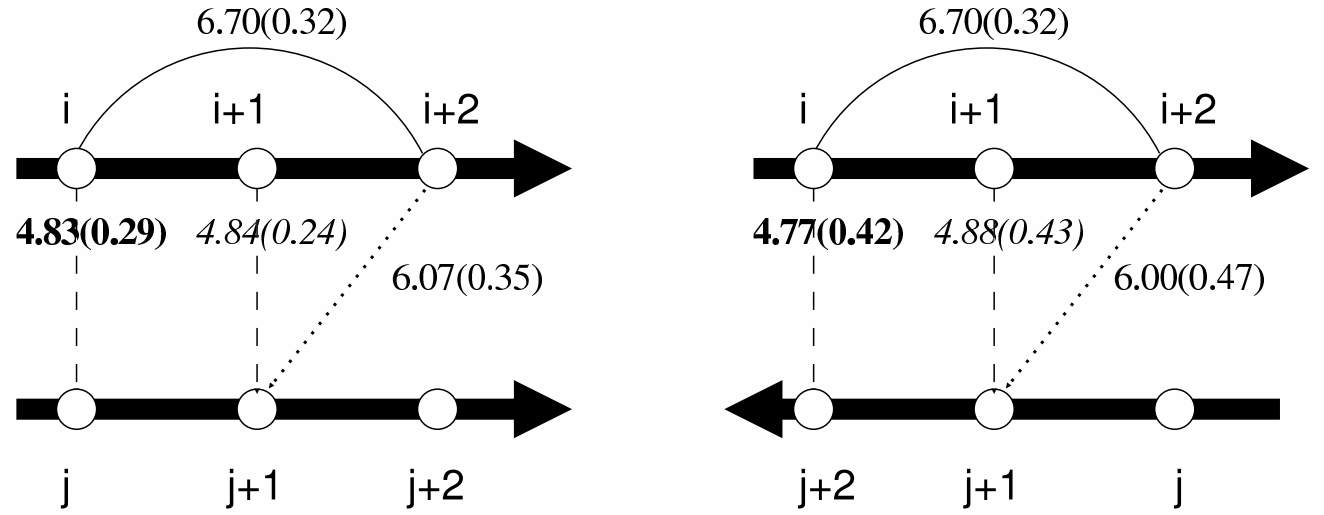

Figure 1
Typical Cα distance in β-sheets. Typical Cα distances computed from the *Ref set* in parallel (left part) and anti-parallel β-sheets. Mean distances are indicated in Å with their standard deviations within parentheses. Separate statistics were computed for distances involving only residues in strand cores (italic) and distances involving residues at strand edges (bold). For the intra-strand distance (type *i* to *i* + 2), no distinction is made on the sheet orientation.

Table 1: Distances in α-helices. *Core*: distances not involving residues at helix edge. *Termini*: distances involving at least one residue at helix edge. Mean distances, computed on the *HRes* set, are indicated in Å with their standard deviations within parentheses.

Type	Core	Termini
i to *i* + 2	5.49(0.20)	5.54(0.25)
i to *i* + 3	5.30(0.64)	5.36(0.39)
i to *i* + 4	6.33(0.71)	
i to *i* + 5	8.72(0.63)	

• *Distance criterion for α-helices* (*C1*). All Cα distances in a sliding window of length w_1 (fixed to 6 in this study) must lie within the interval $[M_\alpha - \varepsilon_H \times SD_\alpha; M_\alpha + \varepsilon_H \times SD_\alpha]$. M_α and SD_α represent the mean and standard deviation of Cα distance distributions in α-helices.

• *Angle criterion for α-helices* (*C2*). All (Φ/Ψ) pairs in a sliding window of length w_2 (fixed to 4 in this study) must satisfy the condition ($\Phi < 0°$ and $-90° < \Psi < 60°$) and one pair at least must fall in the highly populated zone of the population matrix, i.e with density> δ_H.

• *Kinks in α-helices are detected using two criteria.*

- Kink criterion K1 is based on the values of (Φ/Ψ) dihedral angles. A helix is interrupted at residue $j + 1$ if the sum $d_{\Phi/\Psi}(j, j + 1) + d_{\Phi/\Psi}(j + 1, j + 2)$ is greater than $\eta_H \times D^{95}_{\Phi/\Psi}$. $d_{\Phi/\Psi}(j, j + 1)$ is analogous to the root mean square deviation on angular value described by Shuchhardt and coll [44]. It measures the distance between dihedral angle pairs of residues j and $j + 1$ in the Ramachandran map. $D^{95}_{\Phi/\Psi}$ is the 95-percentile of the distribution of such distances.

- Kink criterion K2 relies on axes. An axis is fitted along the helix, by minimizing the function $D_{axis} = \frac{1}{n}\sum_i (d_i - d_m)^2$ with n the number of residues in the helix, d_i the distance from the *ith* Cα to the axis, and d_m the mean of the d_is. For a perfect (linear) helix the value of D_{axis} is zero and the corresponding vector is the axis of the cylinder circumscribed by backbone atoms. A helix is interrupted if it appears better to fit it with two axes. These

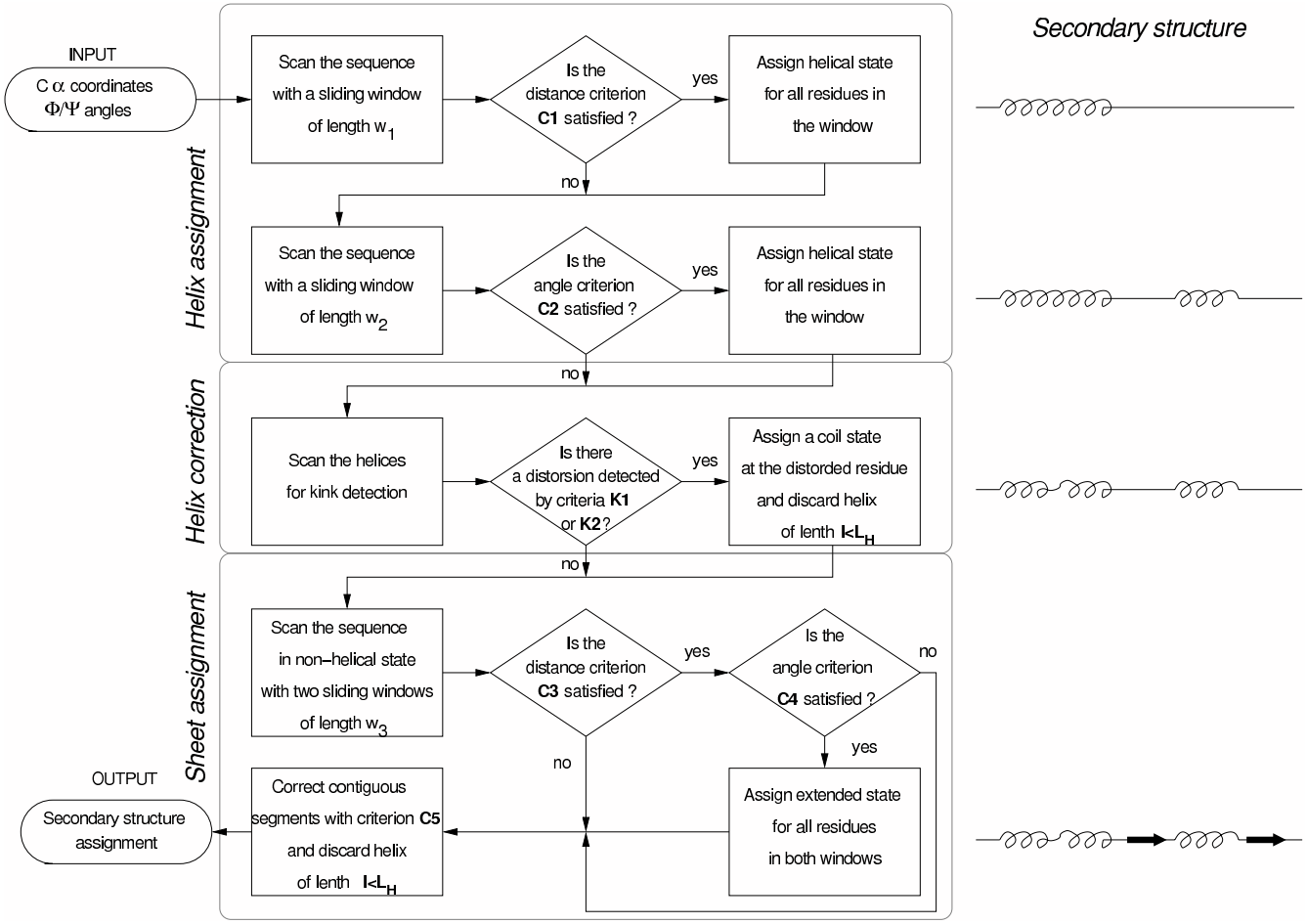

Figure 2
Flow-chart of the kaksi heuristic for secondary structure assignment. Minimum length for helices is set to $L_H = 5$. The criteria C1, C2, C3, C4, C5, K1 and K2 are detailed in the text.

two axes must make an angle greater than θ_k (θ_k fixed to 25° in this study).

• *Distance criterion for β-sheets* (*C3*). All the Cα distances in two sliding windows of length w_3 (here $w_3 = 3$) must be in the interval $[M_\beta - \varepsilon_b \times SD_\beta; M_\beta + \varepsilon_b \times SD_\beta]$. M_β and SD_β represent the mean and standard deviation of Cα distance distributions in β-sheets.

• *Angle criterion for β-sheets* (*C4*). For each (Φ/Ψ) angle pair falling in the populated zone of the Ramachandran map (density > 0), we increment a counter *score*(*sheet*) by 1. If a (Φ/Ψ) angle pair of the central residue of a sliding window verifies -120° < Ψ < 50°, then *score*(*sheet*) is reset to zero. The final *score*(*sheet*) must be greater or equal to σ_b.

• *Contiguous segments correction, criterion* (*C5*). If a helix and a strand are adjacent, a coil is introduced in between, shortening the helix by one residue.

Empirically, the optimal parameter values are: $\varepsilon_H = 1.96$, $\eta_H = 2.25$, $\varepsilon_b = 2.58$ and $\sigma_b = 5$.

Comparative methods for secondary structure assignment and reduction to three states

KAKSI assignments are compared to the assignments given by five available methods on the *Comparison sets:* DSSP [21], STRIDE [23], PSEA [27], XTLSSTR [28] and SECSTR [24]. HELIX and SHEET records in PDB files are also considered as an independent assignment method.

When needed, secondary structure assignments are reduced to three classes (H for α-helix, b for β-strand, c for

coil) as follows: DSSP, STRIDE and SECSTR: (H,G,I) = H, (E,b) = b, others (S,T,blank) = c; XTLSSTR: (G,g,H,h) = H, (E,e) = b, others (T,N,P,p,-) = c. PSEA assigns only three states. XTLSSTR possibly provides several alternative assignments for one residue. In that case, only the first assignment is considered. When dealing with NMR structures, only the first model is analyzed.

Comparison measures

Secondary structure content

The secondary structure content of a dataset is measured by the percentage of residues involved in the three structural classes: α-helix, β-strand and coil.

Overall agreement

The C_3 score is the percentage of residues assigned in the same state when comparing two different assignments: $C_3 = N_{id}/N_{tot}$ with N_{id} the number of residues for which both assignments are identical, and N_{tot} the total number of residues with defined secondary structure. It is analogous to the Q_3 score used to evaluate secondary structure prediction.

Segment based-agreement

• The mean agreement based on secondary structure segments is measured by the percentage of Segment OVerlap (SOV). We use the SOV definition described by Zemla and coworkers [45]. For state i (α-helix, β-strand or coil) the segment overlap measure is defined as:

$$SOV(i) = \frac{1}{N(i)} \sum_{s(i)} \frac{minov(s_1, s_2) + delta(s_1, s_2)}{maxov(s_1, s_2)} \times len(s_1)$$

with the normalization value $N(i)$ defined as:

$$N(i) = \sum_{S(i)} len(s_1) + \sum_{S'(i)} len(s_1).$$

The sums on $S(i)$ are taken over all the segment pairs in state i which overlap by at least one residue. The sum on $S'(i)$ is taken over the remaining segments in state i found in the reference assignment 1, $len(s_1)$ is the number of residues in segment s_1, $minov(s_1, s_2)$ is the length of overlap of s_1 and s_2, $maxov(s_1, s_2)$ is the total extend for which either of the segments S1 and s_2 has a residue in state i, and $delta(s_1, s_2)$ is defined as:

$$min\ \{maxov(s_1, s_2) - minov(s_1, s_2);\ minov(s_1, s_2);\ int(len(s_1)/2);\ int(len(s_2)/2)\},$$

where $min\ \{x1; x2; x3;...; xn\}$ is the minimum of n integers. This formula is usually employed to compare a secondary structure prediction (S_2) with a secondary structure description (S_1) taken as reference. The roles of S_1 and S_2 are thus not symmetrical.

• Length of pair of segments used for the SOV computation are collected. A pair is defined each time there is at least one residue in common between assignment X and Y. Unpaired secondary structure elements are ignored in this analysis. These length pairs can be viewed on a bi-plot (length(X) versus length(Y)).

Helix geometry analysis with an external software

The HELANAL software developed by Kumar and Bansal [33] is dedicated to helix geometry analysis. HELANAL takes as input a PDB file and a description of helix boundaries. It calculates local axes every four residues. The geometry of a helix is determined by the angles between axes and the goodness of fit of the helix trace with a circle or a line. Helices are then classified as kinked (K), linear (L) or curved (C). HELANAL can leave a helix unclassified if its geometry is ambivalent. The minimum length for a helix to be analyzed is nine residues.

In this study, HELANAL is used as an external control of helix geometry. All α-helices in the *comparison sets* are submitted to HELANAL analysis. Different assignment methods are used to provide alternate definition of helices boundaries.

Availability and requirements

• Project name: KAKSI

• Project home page: http://migale.jouy.inra.fr/mig/mig_fr/servlog/kaksi/

• Operating system: Linux

• Programming langage: C

• Other requirements: libxml2 >= 2.6, see ftp://xmlsoft.org/

• License: GNU GPL

• Any restrictions to use by non-academics: no

• Implementation: the software is composed of 2 programs: KAKSI takes a PDB file as input and prints the assigned secondary structure (and other data of intereset) in an XML output K2R reads a KAKSI XML output file and outputs the data in various FASTA format files by default. K2R allows users to easily implement any new output format they whish. a lot of different informations in raw formats (mainly FASTA format).

The source code is available on the project home page.

List of abbreviations used

3D: three-dimensional, Cα: backbone α-carbon, NMR: Nuclear Magnetic Resonance, PDB: Protein Data Bank.

Authors' contributions

JM and AM developped the program. GL carried out the comparison between different assignments. JM GL and JFT carried out the analysis. JM, AdB and JFG conceived the study and participated in its design and coordination

Additional material

Additional File 1

C_3 scores for all datasets.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1472-6807-5-17-S1.pdf]

Additional File 2

Graphical views of C_3 scores for the HRes set.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1472-6807-5-17-S2.pdf]

Additional File 3

SOV scores for all datasets.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1472-6807-5-17-S3.pdf]

Additional File 4

Length of pairs of helices and strands on separate plots.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1472-6807-5-17-S4.pdf]

Additional File 5

Helix geometry analysis on all datasets.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1472-6807-5-17-S5.pdf]

Additional File 6

Urls to retrieve the list of structures used in this study.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1472-6807-5-17-S6.pdf]

Additional File 7

Φ/Ψ repartition in helices and strands defined by the PDB.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1472-6807-5-17-S7.pdf]

Acknowledgements

This research was funded in part by the 'ACI Masse de données'. We are grateful to INRA for awarding a doctoral Fellowship to JM and to the Ministère de l'Education Nationale, de l'Enseignement supérieur et de la Recherche for awarding a doctoral Fellowship to JFT.

References

1. Pauling L, Corey RB, Branson HR: **The structure of proteins; two hydrogen-bonded helical configurations of the polypeptide chain.** *Proc Natl Acad Sci USA* 1951, **37(4):**205-211.
2. Pauling L, Corey RB: **The pleated sheet, a new layer configuration of polypeptide chains.** *Proc Natl Acad SciU S A* 1951, **37(5):**251-256.
3. Gibrat JF, Madej T, Bryant SH: **Surprising similarities in structure comparison.** *Curr Opin Struct Biol* 1996, **6(3):**377-385.
4. Murzin AG, Brenner SE, Hubbard T, Chothia C: **SCOP: a structural classification of proteins database for the investigation of sequences and structures.** *J Mol Biol* 1995, **247(4):**536-40.
5. Orengo CA, Michie AD, Jones S, Jones DT, Swindells MB, Thornton JM: **CATH-a hierarchic classification of protein domain structures.** *Structure* 1997, **5(8):**1093-1108.
6. Sayle RA, Milner-White EJ: **RASMOL: biomolecular graphics for all.** *Trends Biochem Sci* 1995, **20(9):**374.
7. Humphrey W, Dalke A, Schulten K: **VMD: visual molecular dynamics.** *J Mol Graph* 1996, **14:**33-38. 27-28.
8. Sali A, Blundell TL: **Comparative protein modelling by satisfaction of spatial restraints.** *J Mol Biol* 1993, **234(3):**779-815.
9. Bradley P, Chivian D, Meiler J, Misura KM, Rohl CA, Schief W, R Wedemeyer W, Schueler-Furman O, Murphy P, Schonbrun J, Strauss C, Baker D: **Rosetta predictions in CASP5: successes, failures, and prospects for complete automation.** *Proteins* 2003, **53(Suppl 6):**457-468.
10. Pollastri G, Przybylski D, Rost B, Baldi P: **Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles.** *Proteins* 2002, **47(2):**228-235.
11. Petersen TN, Lundegaard C, Nielsen M, Bohr H, Bohr J, Brunak S, Gippert GP, Lund O: **Prediction of protein secondary structure at 80% accuracy.** *Proteins* 2000, **41:**17-20.
12. Jones DT: **Protein secondary structure prediction based on position-specific scoring matrices.** *J Mol Biol* 1999, **292(2):**195-202.
13. Frishman D, Argos P: **The future of protein secondary structure prediction accuracy.** *Fold Des* 1997, **2(3):**159-62.
14. Rost B: **Review: protein secondary structure prediction continues to rise.** *J Struct Biol* 2001, **134(2–3):**204-218.
15. Schulz GE, Barry CD, Friedman J, Chou PY, Fasman GD, Finkelstein AV, Lim VI, Pititsyn OB, Kabat EA, Wu TT, Levitt M, Robson B, Nagano K: **Comparison of predicted and experimentally determined secondary structure of adenyl kinase.** *Nature* 1974, **250(462):**140-2.
16. Robson B, Garnier J: *Introduction to Proteins and Protein Engineering* Amsterdam: Elsevier Press; 1986.
17. Barlow DJ, Thornton JM: **Helix geometry in proteins.** *J Mol Biol* 1988, **201(3):**601-619.
18. Kumar S, Bansal M: **Structural and sequence characteristics of long alpha helices in globular proteins.** *Biophys J* 1996, **71(3):**1574-1586.
19. Kumar S, Bansal M: **Geometrical and sequence characteristics of alpha-helices in globular proteins.** *Biophys J* 1998, **75(4):**1935-1944.
20. Levitt M, Greer J: **Automatic identification of secondary structure in globular proteins.** *J Mol Biol* 1977, **114(2):**181-239.
21. Kabsch W, Sander C: **Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features.** *Biopolymers* 1983, **22(12):**2577-637.
22. Berendsen HJC, van der Spoel D, van Drunen R: **GROMACS: A message-passing parallel molecular dynamics implementation.** *Comp Phys Comm* 1995, **91:**43-56.
23. Frishman D, Argos P: **Knowledge-based protein secondary structure assignment.** *Proteins* 1995, **23(4):**566-579.
24. Fodje MN, Al-Karadaghi S: **Occurrence, conformational features and amino acid propensities for the pi-helix.** *Protein Eng* 2002, **15(5):**353-358.
25. Richards FM, Kundrot CE: **Identification of structural motifs from protein coordinate data: secondary structure and first-level supersecondary structure.** *Proteins* 1988, **3(2):**71-84.

26. Sklenar H, Etchebest C, Lavery R: **Describing protein structure: a general algorithm yielding complete helicoidal parameters and a unique overall axis.** *Proteins* 1989, **6:**46-60.
27. Labesse G, Colloc'h N, Pothier J, Mornon JP: **P-SEA: a new efficient assignment of secondary structure from C alpha trace of proteins.** *Comput Appl Biosci* 1997, **13(3):**291-5.
28. King SM, Johnson WC: **Assigning secondary structure from protein coordinate data.** *Proteins* 1999, **3(35):**313-320.
29. Dupuis F, Sadoc JF, Mornon JP: **Protein secondary structure assignment through Voronoi tessellation.** *Proteins* 2004, **55(3):**519-528.
30. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE: **The Protein Data Bank.** *Nucleic Acids Res* 2000, **28:**235-242.
31. **PDB Format Description Version 2.2** [http://www.rcsb.org/pdb/docs/format/pdbguide2.2/guide2.2_frame.html]
32. Colloc'h N, Etchebest C, Thoreau E, Henrissat B, Mornon JP: **Comparison of three algorithms for the assignment of secondary structure in proteins: the advantages of a consensus assignment.** *Protein Eng* 1993, **6(4):**377-382.
33. Bansal M, Kumar S, Velavan R: **HELANAL: a program to characterize helix geometry in proteins.** *J Biomol Struct Dyn* 2000, **17(5):**811-819.
34. Fourrier L, Benros C, de Brevern AG: **Use of a structural alphabet for analysis of short loops connecting repetitive structures.** *BMC Bioinformatics* 2004, **5:**58.
35. Andersen C, Rost B: **Automated Secondary Structure Assignment.** In *Structural Bioinformatics* Edited by: Bourne PE, Weissig H. Hoboken: Wiley-Liss; 2003:341-363.
36. Brenner SE, Koehl P, Levitt M: **The ASTRAL compendium for protein structure and sequence analysis.** *Nucleic Acids Res* 2000, **28:**254-256.
37. **ASTRAL website** [http://astral.berkeley.edu/]
38. Hobohm U, Scharf M, Schneider R, Sander C: **Selection of a representative set of structures from the Brookhaven Protein Data Bank.** *Protein Science* 1992, **1:**409-417.
39. **WHATHIF website** [http://swift.cmbi.kun.nl/whatif/select/]
40. **PISCES website** [http://dunbrack.fccc.edu/PISCES.php]
41. Wang G, Dunbrack RLJ: **PISCES: a protein sequence culling server.** *Bioinformatics* 2003, **19(12):**1589-1591.
42. **PAPIA website** [http://mbs.cbrc.jp/papia/papia.html]
43. **PDB website** [http://www.rcsb.org/pdb/]
44. Schuchhardt J, Schneider G, Reichelt J, Schomburg D, Wrede P: **Local structural motifs of protein backbones are classified by self-organizing neural networks.** *Protein Eng* 1996, **9(10):**833-842.
45. Zemla A, Venclovas C, Fidelis K, Rost B: **A modified definition of Sov, a segment-based measure for protein secondary structure prediction assessment.** *Proteins* 1999, **34(2):**220-223.
46. Kraulis PJ: **MOLSCRIPT: A Program to Produce Both Detailed and Schematic Plots of Protein Structures.** *J Applied Crystallogr* 1991, **24:**946-950.

Hidden Markov Model for protein secondary structure

Juliette Martin, Jean-Francois Gibrat, and Francois Rodolphe

Unité Mathématique Informatique et Génome,
INRA, Domaine de Vilvert,
78350 Jouy-en-Josas Cedex, France
(e-mail: [Juliette.Martin,Jean-Francois.Gibrat,
Francois.Rodolphe]@jouy.inra.fr)

Abstract. We address the problem of protein secondary structure prediction with Hidden Markov Models. A 21-state model is built using biological knowledge and statistical analysis of sequence motifs in regular secondary structures. Sequence family information is integrated *via* the combination of independent predictions of homologous sequences and a weighting scheme. Prediction accuracy with single sequences reaches 65.3% and raises to 72% of correct classification with profile information.
Keywords: α-helix, β-sheet, prediction.

1 Introduction

Proteins are the main actors of living cells. Many cellular constituents are made out of proteins. Almost all enzymes are proteins, cellular pumps and motors are made out of proteins.

The function of a protein strongly depends of its 3D-structure. For instance, enzymes need to have a tight spatial complementarity with their substrates (reaction partners). Thus knowledge of a protein structure gives relevant clues to its function.

Since genome sequencing started, the even widening gap between the number of protein sequences and protein structures available in databases enhances the utility of structure prediction methods. Because of the structure-function relationship, structures are more conserved than sequences during evolution and therefore different sequences can have the same 3D structure.

Structure prediction methods fall into two categories:

- comparative modeling if a related structure is known and can be used to derive a global model,
- *de novo* prediction if there is no related structure available.

We are presently interested in the latter. *De novo* prediction methods often require a first step of local structure prediction: secondary structure prediction in our case. Three canonical classes of secondary structures are considered : α-helices, β-strands and coil, see figure 1.

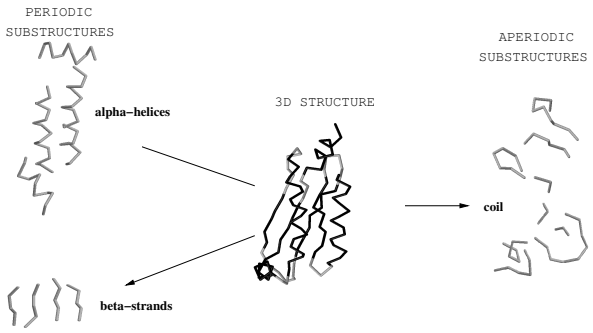

Fig. 1. Secondary structure of proteins. A 3D protein structure (center) can be described in term of secondary structures: α-helices, β-strands (left side) and coil (right side). Only C-α are shown, periodic substructures are indicated in black in the full 3D structure.

α-helices and β-strands are geometrically periodic sub-structures frequently occurring in 3D structures (about 50% of residues in proteins are involved in α-helices and β-strands). Coil denotes all sequence segments which do not fall into one of these two categories.

We use Hidden Markov Models to predict the three classes of secondary structure. The model is built using prior biological knowledge and pattern analysis in protein sequences.

2 Data set

The data set is a subset of 2530 structural domains taken from ASTRAL 1.65 [Brenner *et al.*, 2000], determined by X-ray, with a resolution factor less than 2.25 $\mathring{A}$ and less than 25% sequence identity. Secondary structure definition is given by an assignment method developed in our laboratory (manuscript in preparation) or by STRIDE method [Frishman and Argos, 1995]. 489743 residues have a defined secondary structure in our data set. 2024 sequences, randomly selected, are used in a four-fold cross validation procedure: three quarters of these sequences are used for parameter estimation and one quarter is used for the test. The remaining 506 sequences are used as an *independent* test set. This test set is never used to estimate model parameters. The use of an independent test set allows to check that no bias is introduced during the model design when searching for characteristic motifs in secondary structures (see hereafter). The number of residues with a defined secondary structure are 94790, 101521, 99796 and 99031 in the cross validation subsets and 94605 in the independent test set. The secondary structure contents are similar in all the subsets: about 39% of residues in α-helix, 24% in β-strand and 37% in coil with our assignment and 38%/22%/40% with STRIDE assignment.

3 Hidden Markov Models: application to secondary structure

In a Markovian sequence, the character appearing at position t only depends on the k preceding characters, k being the order of the Markov chain. Hence, a Markov chain is fully defined by the set of probabilities of each character given the past of the sequence in a k-long window: the transition matrix. In the hidden Markov model, the transition matrix can change along the sequence. The choice of the transition matrix is governed by another Markovian process, usually called the *hidden process*. Hidden Markov models are thus particularly useful to represent sequence heterogeneity. These models can be used in predictive approaches: some algorithms like the Viterbi algorithm and the forward-backward procedure allow to recover which transition matrix was used along the observed sequence.

In our case, it is known that different structural classes have different sequence specificity. Intuitively we want to use different Markov chains to model different secondary structures. Figure 2 illustrates the HMM-translation of our secondary structure prediction problem.

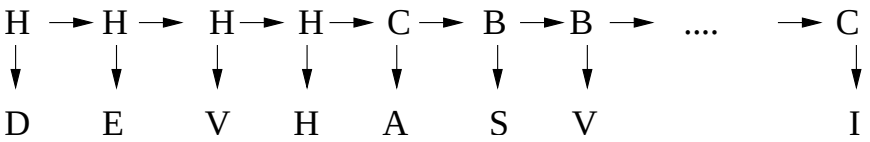

Fig. 2. Secondary structure prediction *via* a hidden Markov model. The upper line represents the secondary structure along a protein sequence: H for a residue in α-helix, B for β-strand, C for coil. The arrows between symbols symbolize the first order dependency of the *hidden process*. The lower line represents the amino-acid sequence of the protein. This is the *observed sequence*. Arrows between the two lines symbolize the dependency between the observed sequence and the hidden chain. The forward/backward algorithm will be used to recover the hidden process from the observed sequence.

The hidden process to be recovered is the secondary structure of the protein. The observed process is the amino-acid sequence. The hidden chain process is a first order Markov chain. Each hidden state is characterized by a distribution of amino-acids. Due to the large alphabet size, the order of the observed chain is 0, which means that amino-acids are independent conditionally on the the hidden process. We use the software called SHOW[1] [Nicolas *et al.*, 2002] to design and train the model and to recover the hidden process. The prediction is achieved with the forward/backward algorithm. Note that this algorithm provides the probability associated to each hidden states at each position.

[1] http://www-mig.jouy.inra.fr/ssb/SHOW/

The simplest model for three-classes prediction is a HMM with three hidden states, each state accounting for a secondary structure class. Parameter estimation of such a model is straightforward because the segmentation is fully determined. But the performance of this model is limited: the Q3 score (proportion of residues with correct prediction) is 58.3%. A random prediction gives a Q3 score equals to 34.5%.

We thus want to design a model that takes into account the specific features of secondary structures.

4 Model of α-helices

A well-characterized sequence motif in α-helices is the amphiphilic motif, i.e., a succession of two polar residues and two apolar residues. This motif occurs when an helix has a side facing the solvent (thus preferentially supporting polar residues) while the other side faces the core of the protein (preferentially supporting apolar residues). This motif is very frequent. With the amino-acids classification; A,V,L,I,F,M,W,C=hydrophobic (h), S,T,Y,N,Q,-H,P,D,E,K,R=polar (p), the motif hhpphh or pphhpp is found in 24% of the helices in our cross-validation set. Glycine (G) residues do not exhibit strong preference for either polar or apolar environment. It is thus considered as a special type of residue and left apart. When reduced to hhpp or pphh, the motif is found in 69% helices. Figure 3 shows the model we propose to take into account the amphiphilic nature of α-helices. States H5 and H6 help to fit the periodicity of an α-helix which is 3.6 residues.

States with hydrophobic preference favour amino-acids A, V, L, I, F, P and M. States with polar preference favour S, T, N, Q, H, D, E, K and R.

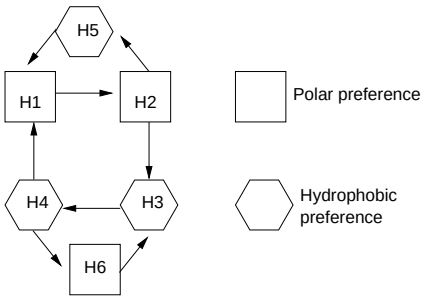

Fig. 3. Model for amphipatic helices

5 Model of β-strands

There is no strong motif characterizing β-strands similar to the amphipatic motif for α-helices. Characteristic motifs are found using a statistical ap-

proach based on exceptional words. A word is over (resp. under)-represented if its frequency in the data is significantly greater (resp. lower) than its expected frequency under some Markovian model. The R'MES software[2] [Bouvier *et al.*, 1999] is dedicated to this task. Amino-acids are grouped as before, the G is put into the hydrophobic group. Sequences of β-strands and α-helices in the cross-validation set are analyzed with R'MES using the Gaussian approximation.

Because the HMM uses a zero order for the observed chain, exceptional words when compared to a zero order Markov model are interesting. Interesting words should also be frequent in absolute (over-represented words are not necessarily frequent) and must not be over-represented in α-helices. We also consider some frequent words, although not over-represented, if they are under-represented in α-helices. Table 1 contains interesting words found in β-strand with R'MES. The over-representation is assessed by R'MES. The relative abundance is evaluated by looking at rank of the word when sorted according to the frequency.

Motif	Occurrence in β-strands	Occurrence in α-helices
hphp	over-represented and frequent	under-represented and not frequent
phph	over-represented and frequent	under-represented and not frequent
pphhh	over-represented and very frequent	under-represented and not frequent
pphph	over-represented and very frequent	under-represented and not frequent
hhhhp	not over-represented, but very frequent	under-represented and not frequent
phhhhp	not over-represented but very frequent	under-represented

Table 1. Interesting motifs in β-strands

Figure 4 shows the model we propose to take into account these words in β-strands. Words hphp and phphp are favoured by the alternation between states b1 and b2. This alternation corresponds to the case of β-strands at the solvent interface with one side facing the solvent and one side facing the core of the protein. The transition from state b4 to itself favours long runs of hydrophobic amino-acids in words pphhh, hhhhp, phhhhp. Long runs of hydrophobic residues are seen when β-strands are buried in the core of proteins. The transition between b2 and b3 favours the apparition of two polar amino-acids surrounded by hydrophobic ones appearing in words pphhh and pphph.

Note that the study of exceptional words on α-helices reveals that the motifs occurring in amphipatic α-helices are over-represented.

[2] http://www-mig.jouy.inra.fr/ssb/rmes/

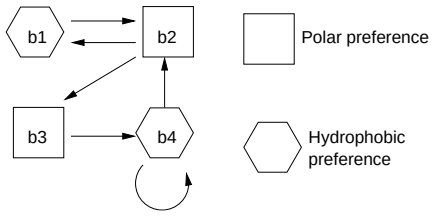

Fig. 4. Model for β-strands

6 Complete HMM for secondary structures

Models of β-strands and α-helices are merged to form a full model of secondary structures.

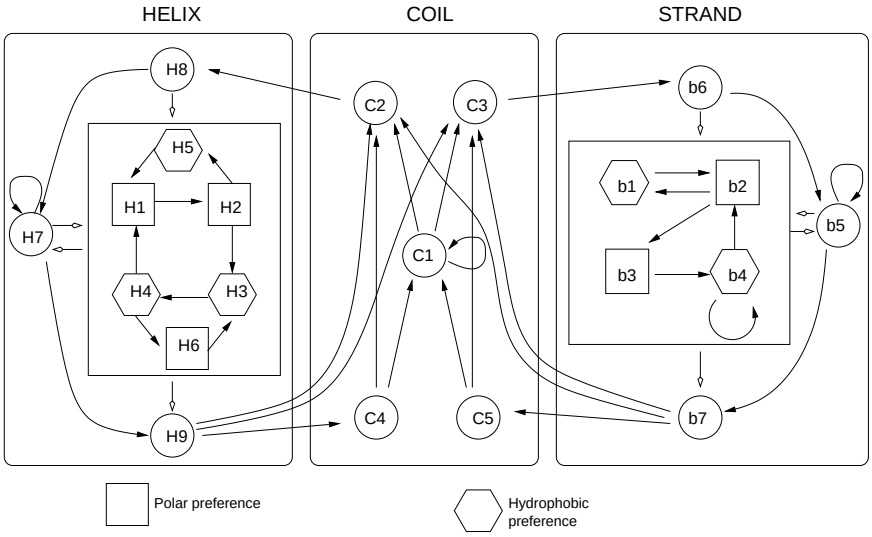

Fig. 5. Full model for secondary structure

Figure 5 shows the full model. Models of α-helices and β-strands integrate informations about frequent/over-represented words, but they don't necessarily reflect the totality of motifs in periodic structures. To allow the presence of β-strands and α-helices that do not fit well in the constrained models, two "generic" states were added (H7, b5). These states show no prior preference for polar or hydrophobic amino-acids. Transitions are allowed between all states of the constrained models and the "generic states". Specific states are added at secondary structure ends (H8, H9, b6, b7), as it is known that there are specific signals such as helix-caps. The coil is not well characterized yet, except the states preceding and following regular secondary structures.

Initial parameters for estimation by the EM algorithm are set as follows:

- Initial transition probabilities are set to $\frac{1}{n}$, with n the number of outgoing states.
- Initial emission probabilities are derived from those obtained on a simple 3-states model. Emission probabilities are manually modified to favour the apparition of polar amino-acids and penalize the emission of hydrophobic amino-acids in polar-preferring states (and vice-versa). No such bias is introduced in other states.

Prediction of the three structural classes (α-helix, β-strand, coil) is achieved by the forward-backward algorithm. The predicted structure is the one with the greatest posterior probability.

7 Integrating information from homologous sequences in the prediction

Protein structures are more conserved than sequences during evolution. Thus different sequences can have the same structure. This information has been successfully used in secondary structure prediction methods [Rost, 2003]. To integrate this information, the prediction is done independently on each sequence of a family. These sequences are detected using a search with PSI-BLAST against a database were the redundancy is reduced to 80% sequence identity. This search generates an average number of 60 sequences per family. Independent predictions are combined with a weighting scheme to generate a prediction for the sequence family using the formula

$$P(state = S/family) = \sum_i \lambda_i \times P(state = S/sequence_i)$$

with $P(state = S/sequence_i)$ provided by the forward-backward procedure and λ_i the weight of sequence i in the family. Sequence weights are computed as proposed in Henikoff and Henikoff [Henikoff and Henikoff, 1994].

Prediction on single sequence provides an accuracy of 65.2% residues correctly classified when compared to our secondary structure assignment, on the cross-validation test set. This score is 65.3% on the learning set and 65.6% on the independent test set. When compared with stride assignment, the accuracy is around 66.3% for all data sets. Hence, we experienced no over-fitting on the training data.

With the family sequence information, the percentage of correct prediction is in the range 71.3 to 72%. Best available methods, that also use sequence families, have achieved accuracy in the range of 78% (reported for reasonnably big datasets on the continuous evaluation server EVA, [Koh *et al.*, 2003]). Thus our results are not fully satisfying yet. However we think that our approach is promising because our model is relatively small,

statistically speaking: the number of independent parameters is only 471. Most of existing methods use neural networks. The number of parameters, when reported, seems to be of the order of thousands [Pollastri *et al.*, 2002]. Moreover, the graphical nature of hidden Markov models allows intuitive data modeling. Along this line, an important perspective of this work is to introduce a geometrical description of coil. The coil class represents about 50% of residues in proteins. Even a perfect three state prediction would leave half of the data with no structural clue. We also think that the sequence family information could be taken into account more efficiently that it is done here. This is another of our perspectives.

References

[Bouvier *et al.*, 1999]A. Bouvier, F. Gélis, and S. Schbath. *RMES : Programs to Find Words with Unexpected Frequencies in DNA Sequences, User Guide (in french)*, 1999.

[Brenner *et al.*, 2000]S.E. Brenner, P. Koehl, and M. Levitt. The astral compendium for protein structure and sequence analysis. *Nucleic Acids Res*, 28(1):254–6, Jan 2000.

[Frishman and Argos, 1995]D. Frishman and P. Argos. Knowledge-based protein secondary structure assignment. *Proteins*, 23(4):566–79, Dec 1995.

[Henikoff and Henikoff, 1994]S. Henikoff and JG. Henikoff. Position-based sequence weights. *J Mol Biol*, 243(4):574–8, Nov 1994.

[Koh *et al.*, 2003]I.Y. Koh, V.A. Eyrich, M.A. Marti-Renom, D. Przybylski, M.S. Madhusudhan, N. Eswar, O. Grana, F. Pazos, A. Valencia, A. Sali, and B. Rost. Eva: evaluation of protein structure prediction servers. *Nucleic Acids Res*, 31(13):3311–5, Jul 2003.

[Nicolas *et al.*, 2002]P. Nicolas, A.S. Tocquet, and F. Muri-Majoube. *SHOW : Structured HOmogeneities Watcher. User Guide*, 2002.

[Pollastri *et al.*, 2002]G. Pollastri, D. Przybylski, B. Rost, and P. Baldi. Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. *Proteins*, 2002.

[Rost, 2003]B. Rost. Prediction in 1d: secondary structure, membrane helices, and accessibility. *Methods Biochem Anal*, 44:559–87, 2003.

How to choose the optimal Hidden Markov Model for secondary structure prediction

Juliette Martin, Jean-François Gibrat, François Rodolphe

INRA, Unité Mathématiques Informatique et Génome, Domaine de Vilvert, 78352 Jouy en Josas Cedex, France.

Abstract

We address the problem of model selection in the case of hidden Markov models dedicated to secondary structure prediction from protein sequence. We propose models with several hidden states to represent each structural class, analogous to labeled HMMs. The problem is to choose the model that achieves the best compromise between the number of parameters and the prediction accuracy. We first train HMMs with an equal number of hidden states to model each structural class: from 1 to 25 states. We show that objective prediction scores, such as Q_3 and SOV, agree well with the BIC criterion. Then we investigate the contribution of each structural class modeling on the global prediction accuracy : a given structural is modeled by an increasing number of states and the other classes are modeled by only one state. We show that the impact on the global accuracy is different for each class. We finally select the optimal model with the BIC criterion. This optimal model has a limited number of hidden states : 15 for α-helices, 9 for β-strands and 12 for coils. Its topology reveals some features of secondary structures and the performance are good for a model working with single sequences. The Q_3 achieves 68 % and the SOV 64.7%.

Keywords : protein, secondary structure prediction, HMM, model selection.

Introduction

Proteins are major constituents of living cells. Many cellular components and the majority of enzymes are proteins. The knowledge of protein 3D structures is fundamental for meaningful biological mechanism understanding. Genome sequencing

projects have generated a huge number of protein sequences. However the experimental determination of protein structures is still a long and difficult task. That is why a lot of researchers around the world are developing structure prediction methods[1].

The 3D structure of a protein is determined by its sequence. During evolution, mutations occur at the DNA level. They are kept as long as the protein encoded by the gene keeps its function and in turn its 3D structure. In consequence, different protein sequences can share a similar structure. This property has been successfully exploited by methods called *comparative modeling methods*. In comparative modeling, a related structure is used to derive the 3D structure of a target sequence. Other methods, *de novo* methods, aim at predicting structures when no related structure is available. In *de novo* prediction, a local structure prediction is often essential to reduce the search space.

Local structure of proteins can be described at different levels. Some groups have developed structural alphabets, which are collections of short structural fragments. These fragments can be used as a Lego blocks, that can be used to rebuild global structures. The most widely-used local description of proteins remains based on secondary structure classification. Secondary structures are local folds commonly seen in proteins. Classically, three classes are distinguished : the α-helix, the β-strand and the coil. Unlike the first two classes, which are periodic motifs, characterized by geometrical features, the coil class is a default description: any residue in non-α and non-β conformation is by definition classified as coil. Secondary structures can be seen as a special kind of structural alphabet, but we should keep in mind that the coil class is poorly characterized.

Many methods have been developed to predict the secondary structure of proteins based solely on their sequences. Most of them use neural networks and have reached a good level of accuracy [2]. But neural networks are not easy to interpret: it is hard to know which features of the local structure have been caught by the model. Hidden Markov Models (HMMs) have been used by Bystroff et al : they built models starting from a library of sequence patterns with conserved structures. These models take into account the transitions observed in 3D structures. They are used in predictive approach [3] .

Our goal is to find an optimal hidden Markov model [4] to perform the classification of residues into the three secondary structure classes. HMMs provide a probabilistic framework for sequence treatment and produce models that are interpretable. Optimality means for us that we want to find the most parsimonious model that performs well, i.e., a model with a reasonable number of parameters compared to the data set it is fitting to.

HMM framework for the secondary structure prediction task

In a Markovian sequence $X_1X_2X_3X_4X_5....X_n$, the occurrence of X_t only depends on the k preceding observations where k is the order of the Markov chain. A Markov chain is defined by a set probabilities of observing each given symbol, given the past in a window of length k: the transition matrix. This model supposes the sequence homogeneous. If we consider protein sequences it means that the amino-acid distribution is the same along the sequence. The different class of secondary structure do not share the same sequence propensities, that is why we need a HMM that allows heterogeneity.

In a HMM, the transition matrix will change along the sequence. The process that governs the choice of Markovian matrices is itself a Markov chain. It is usually called the hidden process, because in predictive uses of HMM it is not observed. The hidden process is a succession of hidden states characterized by their transition matrices. The HMM formalism provides algorithms to recover the optimal hidden sequence when only the observed process is known.

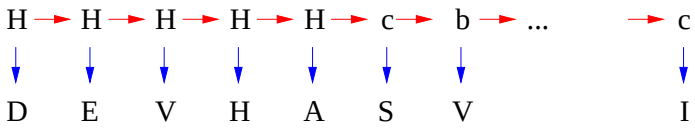

Figure 1: Translating the structure prediction problem into a HMM. The hidden process (upper line) is the succession of secondary structures: H for α-helix, b for β-strand, c for coil. The observed sequence (lower line) is the amino-acid sequence. Red arrows symbolize the first order dependence of the hidden process. Blue arrows indicate the dependence between the observed sequence and the hidden process. Successive amino-acids are independent conditionally to the hidden process.

Figure 1 illustrates the HMM framework for secondary structure prediction. In our case, the hidden process to be recovered will be the secondary structure of the protein and the observation, the amino-acid sequence. Due to the large amino-acid alphabet cardinal, we will only consider models with a first order Markov chain for the hidden process and a zero order Markov chain for the observed process. We use the forward-backward algorithm [4] to predict the structure with the maximum probability at each position.

A very basic model

A very simple HMM to perform our three-states classification is a HMM with three states, one for each class we want to predict. This model is shown Figure 2.

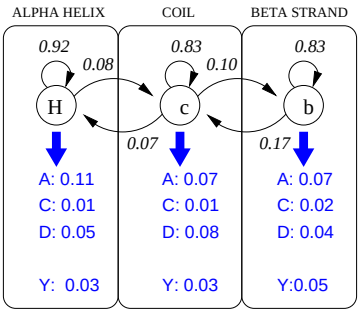

Figure 2: 3-states HMM for secondary structure prediction. The probabilities associated to each states are in italic. Amino-acid distributions of each state are in blue.

Parameters are easily estimated from the dataset, by counting the transitions between secondary structures and the amino-acid composition of each structural class.

In this work we use a dataset of 2530 protein structures from the ASTRAL database 1.65 (`http://astral.berkeley.edu/`). There is less than 25% identity between sequences. HMMs are handled with SHOW (`http://www-mig.jouy.inra.fr/ssb/SHOW/show-20040126.tar.gz`).

We train and select our HMMs with a four-fold cross-validation procedure on 2024 structures: this is the cross-validation set. Three quarters of the cross-validation set are used to estimate the model parameters, this set is called the *cross-va learning set*, and the remaining quarter is used for testing the performance, this is the *cross-va test set*. This procedure is repeated four times with non-overlapping *cross-va test sets*. The prediction accuracy reported are the mean values obtained on the four models.

One fifth of the data (506 structures) are kept apart to assess the performance of the model. It is never used in model selection, only for testing the prediction on unseen data. This is the *independent test set*.

Not surprisingly, the three-states model does not perform very-well in prediction. The Q_3 score, defined by $Q_3 = \frac{N_{good}}{N_{tot}} \times 100$, where N_{good} is the number of positions with correct prediction and N_{tot} the total number of positions, is only 58.5% (vs 34% for a random prediction). In such a model, the assumption is that each class of secondary structure can be described by one mean behavior. Obvi-

ously, this assumption is too restrictive and does not account for the complexity of secondary structures.

Toward more complex models

Bigger models are needed to incorporate more information about the architecture of secondary structures. If we describe our HMM by three boxes, one for α-helices, one for β-strands and one for coil (the rounded rectangles in Figure 2), a logical idea is to put more states in each box and the question becomes "How do we choose the appropriate size of the three boxes ?". The problem we are facing is a *model selection problem*. One solution would be to use prior biological knowledge to choose the appropriate number of states in each box. In this paper we examine the alternative solution that consists in choosing the optimal box sizes using statistical and performance criteria.

Estimation of the model parameters

The estimation procedure will be different here, since our data are labeled as helix, strand and coil, without further indication. In other words, we know in which box is a residue, but not in which particular state of the box. The well-known EM algorithm allows parameter estimation in an unsupervised framework. Here we are in a semi-supervised framework. This EM algorithm can easily be applied to our problem if we consider that the models we are building are equivalent to models that would emit simultaneously two sequences : on the one hand the secondary structures, on the other hand, the protein sequence. This is the type of HMM called "labeled HMM" described by Krogh [5]. The Figure 3 illustrates this model.

To estimate transitions and amino-acid emission parameters, we assume the "structure" part of the model known with probabilities equal to 0 or 1, and we allow the re-estimation procedure only for amino-acid emissions and transition.

A uniformly growing model

In a first attempt, we assume that each class can be described by the same number of states. In other words, all the boxes have equal size. We design models with 1 to 75 states per box, estimate the parameters with the EM algorithm, and perform the prediction with the forward-backward procedure. The EM algorithm is a deterministic iterative algorithm: the optimum reached depends on the starting point. Thus it is sensitive to the problem of local maxima. Here, 10 starting points are

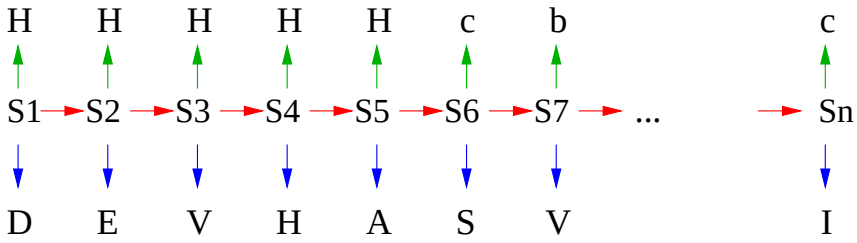

Figure 3: Double-track HMM for parameters estimation. The structure part (green) is kept fixed with emission probabilities equal to 0 or 1. Now, a structural class corresponds to a set of states. Transitions (red) and emission probabilities (blue) are estimated with the EM algorithm. The structure part is discarded from the model definition during the predictive step.

tested with different random probabilities for amino-acid emissions, and the model with the best likelihood is selected. For models with 10 to 25 states per box, we also tested 100 starting points. Initial transition probabilities are uniform.

Figures 4, 5 and 6 show the evolution of prediction scores when the model is growing. We computed the Q3 score, the SOV score and the Information Gain. The SOV score, defined by Zemla and co-workers [6] is a measure of how well the segments of secondary structure overlap . The Information Gain has been used by Karchin and co-workers to compare the predictability of structural alphabets with different sizes [7]. It is defined by

$$IG = \frac{1}{N} \sum \log_2 \frac{P_{pred}(observedstructure)}{P_{\emptyset}(observedstructure)}$$

where N is the size of the dataset, $P_{pred}(observedstructure)$ is the probability of the observed secondary structure given by the HMM and $P_{\emptyset}(observedstructure)$ is the background probability of the observed secondary structure, i.e., the frequency in the dataset.

We report:

- the mean scores on the *cross-va learning set*, i.e., the performance on the data used for parameter estimation, with the mean taken over the four models,
- the mean scores on the *cross-va test set*, i.e., the performance on the data not used for parameter estimation, with the mean taken over the four models,
- the mean scores on the *independent test set*, with the mean taken over the four models.

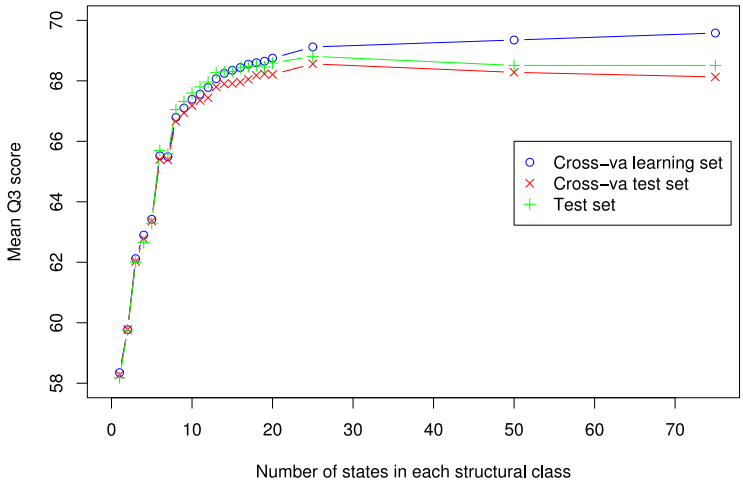

Figure 4: Q3 score evolution

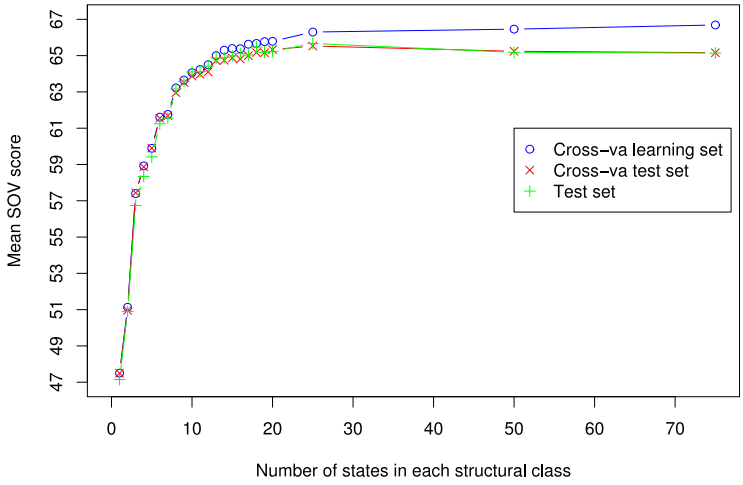

Figure 5: SOV score evolution

The evolution of prediction scores shows that a good level of performance is reached with 10-15 states per box. Adding more states has little effect on the Q_3 score. With very big models, we even observe an over-fitting effect: the Q_3 score is 69.6 % on the *cross-va learning set* and 68.1% only on the *cross-va test set* with a model with 75 states per box.

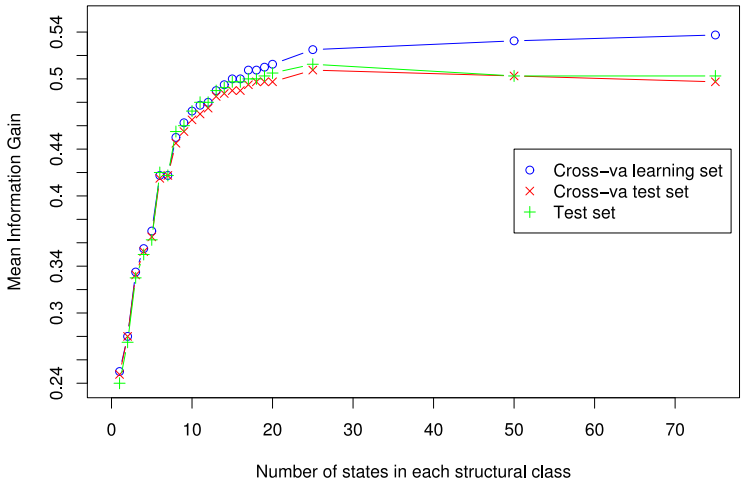

Figure 6: Information Gain evolution

To ensure that the stationary level we observe for the performance is not due to the problem of local maxima, we tried 100 starting points for models with 10 to 25 states per box. Figure 7 shows the evolution of the Q3 score for these models.

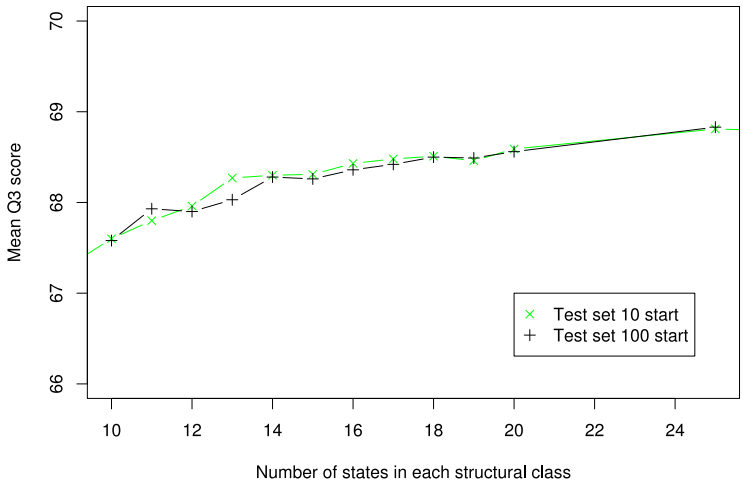

Figure 7: Q3 score evolution

The performance is equivalent whatever the number the number of starting

points, showing that this is not a problem of local maximum. In fact, models with too many parameters will capture some specific features of the learning set and will perform poorly on new data. We also see that it not easy to decide when it is worth to add some states.

Working in a semi-supervised framework, we can use the BIC criterion usually employed in unsupervised learning. The BIC criterion is a penalized likelihood criterion: $BIC = logL - 0.5 \times k \times log(N)$, where $logL$ is the log-likelihood of the learning set, k is the number of free parameters and N is the amount of data used to estimate the parameters (here the number of positions with both sequence and structure available). As the size of the model increases, the training set is better represented by the model and then the likelihood increases as well. The number of parameters of the model also increases. The BIC criterion aims at finding the optimal balance between fitting to the data and having enough data to provide a correct parameter estimation. Figure 8 shows the evolution of BIC in our four data sets as a function of the model size. Maximum BIC is obtained for comparable values in the four datasets: 13 or 14 states. Thus, we observe a good agreement between the BIC criterion and the performance scores.

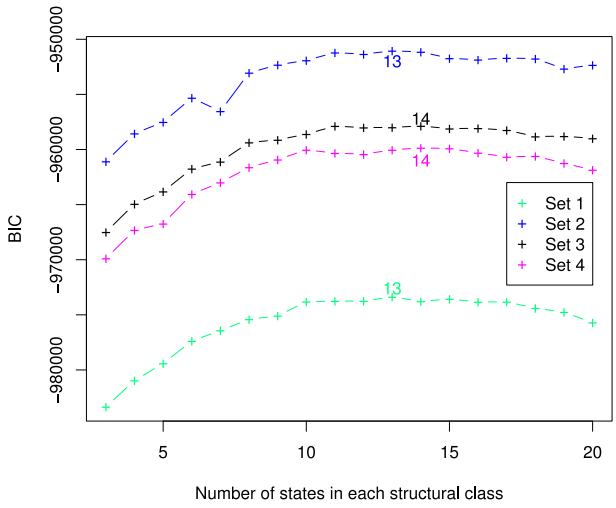

Figure 8: BIC evolution when the HMM is growing.

Now we would like to compare the different models. To assess the similarity of models, we use the measure described by Rabiner [4]. The symmetric distance between two models M1 and M2 is defined by:

$$D_s(M_1, M_2) = \frac{D(M1, M2) + D(M2, M1)}{2}$$

with $D(M1,M2)$ the non-symmetric distance between M_1 and M_2, defined by:

$$D(M1,M2) = \frac{1}{T}|logP(O^{(2)}|M_1) - logP(O^{(2)}|M_2)|$$

where $O^{(2)}$ is a sequence of length T generated by model M_2 and $logP(O^{(2)}|M_1)$ denotes the log-likelihood of the sequence under the model M_1.

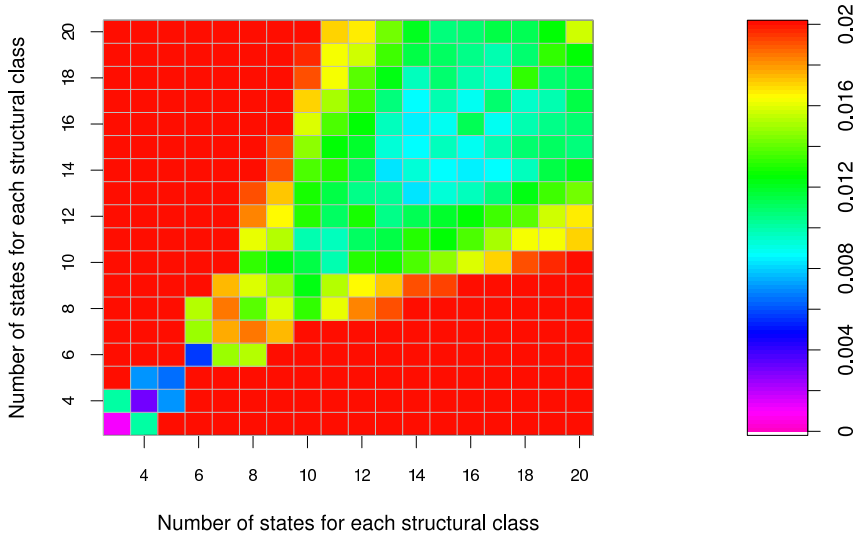

Figure 9: Distances between models

We compute two kinds of distances between HMMs:

- The intra-model distance is the distance between two models with the same number of states but estimated on different data sets. Since we have 4 different overlapping learning sets, the intra-model distance is defined as the mean of the 6 distances computed between any two models.
- The inter-model distance is the mean of the distances between models with different number of states but estimated on the same learning set.

All these distances are plotted on Figure 9. As expected, we obtain low values on the diagonal, showing that models of same size estimated on different data sets are equivalent. A “stability area” is reached for models with 13-17 states per box:

distances between these states are low, indicating that these models are roughly statistically equivalent.

Thus, by surveying the prediction accuracy (Q_3 score), a statistical criterion (BIC) and the model similarities (Rabiner distances), we come to the conclusion that a reasonable -in the terms of number of parameters- yet good performing model should not have more than about 15 states for each structure.

In this first approach, we made the assumption that each class of secondary structure can be modeled by the same number of hidden states. This is a strong hypothesis. About 30% of residues in proteins are involved in α-helices, 20% in β-sheets and the remaining 50% in coil. Solely because of this misbalanced distribution, it is intuitively conceivable that some classes will "saturate" faster than others. Moreover, it is hard to have an idea of the intrinsic complexity of each secondary structure, in term of number of hidden states.

Each box growing

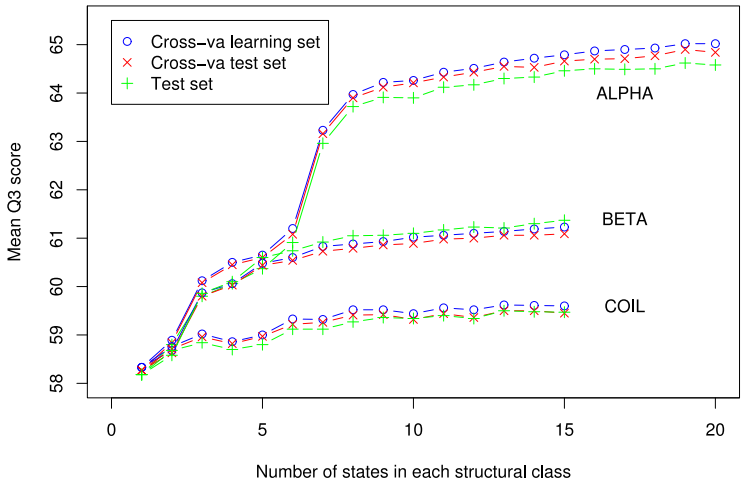

Figure 10: Q3 score evolution when the HMM is growing.

We now let the model grow in one box, keeping only one state in the two other boxes. In Figure 10 we show the evolution of the Q3 score when each sub-model is growing. It is striking that the contribution of α-helices to the global prediction accuracy is greater than the other boxes. A straightforward explanation is that α-helices are more abundant than β-sheets. However, it is surprising that the increase

of the coil box results in the smallest Q3 increase, since it is the most represented class. The reason could be that the coil class is a default class, encompassing all non-α and non-β residues. In consequence, a variety of different features are mixed in this class.

The BIC criterion was also tested on these models, to determine the range of saturation for each class. We found that the maximum BIC was obtained for about 15 states for α-helices, 8 states for β-strands and 9 for coils.

Choice of the optimal combination of box sizes

The optimal combination of box sizes is not necessarily the combination of optimal box sizes independently determined for each structure. Indeed, the preceding approach neglects the fact that the different classes probably have complex connectivity between them, so adding a state in one box can influence the neighboring boxes.

We tried all the models with 12 to 16 states in the α-helix box, 6 to 10 in the β-strand box and 5 to 13 in the coil box, and computed the BIC for each model. The maximum BIC was obtained with 15 states for α-helices, 9 for β-strands and 12 for coils.

Structure of the optimal model

As HMMs allow data interpretation, it is interesting to have a look at the structure of the states graph. All the transitions between α box and coil box, between β box and coil box and within each box are initially allowed. At the end of parameter estimation, a number of them are null. For example, 72 transitions internal to the α box remain, on the 225 initially allowed (32% of initial transitions). 54% of the internal transitions remains in the β box and 61% within the coil box.

In Figure 11 we show the main transitions between hidden states in the optimal model. For clarity, we only indicate transitions associated to probabilities greater than 0.1.

The topologies found for different class of secondary structures are fairly different from each other. The helix architecture shows a linear structure with two cycles of 4 and 3 states. There are very few strong self transitions. On the opposite, the β architecture is more complex, with self-transitions and a highly connected structure. In these two boxes, we notice that there are alternative starting and ending states that are connected to each other. For example, strong transitions occur from the coil box to states H10, H9, H7, H3, and H12. These states further lead to the cyclic core of the helix topology. Thus there are several alternative sequence

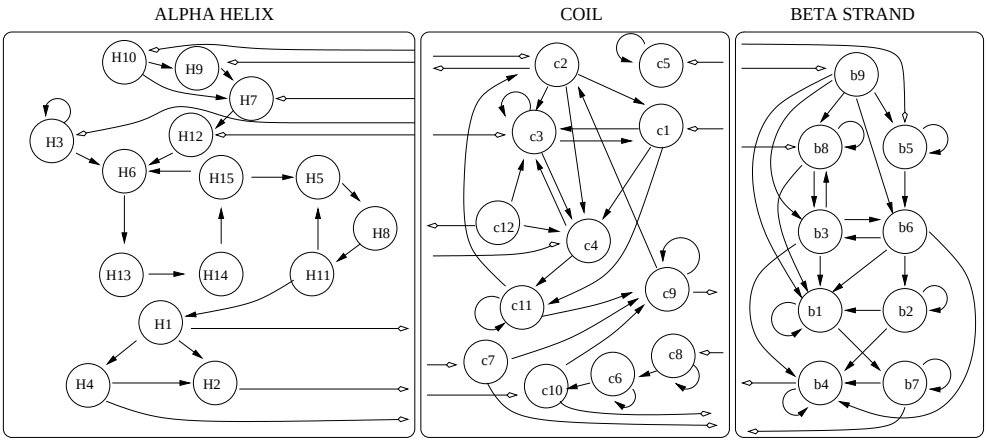

Figure 11: Topology found by the optimal HMM. For clarity, only transition with probability greater than 0.1 are shown. Plain arrows indicate intra-box transitions and open arrows indicate inter-box transitions.

signals for helix and strand edges. For exemple, an amphipathic helix can start on either the hydrophilic or the hydrophobic side. The architecture of the coil class is also quite complex, as expected for a mixture of all non-periodic local structures.

Prediction accuracy of the optimal model

Table I: Performance of the optimal model

	Kaksi			Stride			DSSP		
	c-v L[a]	c-v T[b]	T[c]	c-v L	c-v T	T	c-v L	c-v T	T
Q3[d]	68.1	67.8	68.0	67.6	67.3	67.5	66.6	66.4	66.4
SOV	65.0	64.8	64.7	64.7	64.6	64.4	63.9	63.7	63.4
IG[e]	0.493	0.483	0.490	0.470	0.460	0.460	0.453	0.448	0.445

[a]Mean performance on the 4 *cross-va learning sets*, i.e., computed on the data used for parameter estimation.

[b]Mean performance on the 4 *cross-va test sets*, i.e., computed on the data *not* used for paramater estimation.

[c]Mean performance on the *independent test set* taken over the 4 different models.

[d]Q3 and SOV are expressed in %.

[e]Information Gain is expressed in bit per residue.

We computed the Q3, SOV and Information Gain on the secondary structure predicted by the optimal model. This model was trained and tested with secondary structure assignment with kaksi, an in-house program (to be published). We also report, in Table I the prediction scores with stride [8] and dssp [9] assignment taken as reference. Kaksi remains used for training. We obtain a Q3 about 68%, which is a good level of accuracy for a model working on single sequences [10]. Furthermore, the scores computed on the different data sets shows that there is no over-fitting toward the data used for estimation. The performance is slightly lower for results obtained with stride and dssp assignments. As the secondary structure contents are similar whether we use kaksi, stride, or dssp we suggest that the difference is due to the fact that model parameters were estimated with kaksi assignment.

The best methods evaluated on the EVA website (`http://cubic.bioc.columbia.edu/eva/` achieve Q3 scores up to 78% on sufficiently large data sets. Unlike our HMM, these methods use multiple sequence alignment. The top-performing methods -e.g. PROFSEC (Rost), PSIPRED (Jones), SAMT99sec (Karplus et al), SSPRO (Pollastri et al), JPred (Cuff and Barton), PROFKing (Ouali and King)- show statistically similar results. The PSIPRED method achieves a Q3 of 78.2% on our data.

Conclusion

Using a BIC criteria, we were able to choose an optimal HMM for secondary structure prediction. We previously showed that the BIC criterion is in agreement with the prediction level and the model similarity measures. This model has 36 hidden states and offers the best compromise between prediction accuracy and a reasonable number of parameters. We are currently working on the integration of multiple alignment information to improve the performance.

References

[1] C. Venclovas, A. Zemla, K. Fidelis, and J. Moult. Assessment of progress over the casp experiments. *Proteins*, 53(Suppl 6):585–95, 2003.

[2] B. Rost. Prediction in 1d: secondary structure, membrane helices, and accessibility. *Methods Biochem. Anal*, 44:559–587, 2003.

[3] C. Bystroff, V. Thorsson, and D. Baker. Hmmstr: a hidden markov model for local sequence-structure correlations in proteins. *J Mol Biol*, 301(1):173–90, 2000.

[4] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, 1989.

[5] A. Krogh. Hidden markov models for labeled sequences. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition.*, pages 140–44, Los Alamitos, California, Oct 1994. IEEE Computer Society Press.

[6] A. Zemla, C. Venclovas, K. Fidelis, and B. Rost. A modified definition of sov, a segment-based measure for protein secondary structure prediction assessment. *Proteins*, 34(2):220–3, 1999.

[7] R. Karchin, M. Cline, Y. Mandel-Gutfreund, and K. Karplus. Hidden markov models that use predicted local structure for fold recognition: alphabets of backbone geometry. *Proteins*, 51(4):504–14, 2003.

[8] D. Frishman and P. Argos. Knowledge-based protein secondary structure assignment. *Proteins*, 23(4):566–79, 1995.

[9] W. Kabsch and C. Sander. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–637, 1983.

[10] G. E. Crooks and S. E. Brenner. Protein secondary structure: entropy, correlations and prediction. *Bioinformatics*, 2004.

Résumé

Les méthodes de novo de prédiction de la structure tridimensionnelle des protéines d'après leur séquence sont utilisées dans les cas les plus difficiles de prédiction. Elles nécessitent souvent une première étape de prédiction de la structure à un niveau local. Les structures secondaires (hélices, feuillets, boucles) constituent un premier niveau de description de la structure locale. Pour réaliser cette prédiction de structure locale, nous proposons une approche utilisant des modèles de chaînes de Markov cachées. Ces modèles mathématiques, outre leur cadre théorique solide, ont l'avantage de permettre une modélisation explicite des données.

Une nouvelle méthode d'assignation automatique des structures secondaires d'après la structure tridimensionnelle est tout d'abord proposée. Le problème de la prédiction des structures secondaires est tout d'abord envisagé par l'utilisation de modèles à trois états cachés, avec différents schémas de prise en compte de la mémoire de la séquence. Des modèles ayant un nombre d'états plus élevé sont proposés selon deux stratégies : construction "experte" d'après les connaissances disponibles sur l'organisation des structures secondaires, ou bien sélection de modèles d'après des critères statistiques et de performance. Une description géométrique des boucles en terme de zones d'angles dièdres est ajoutée, pour fournir une prédiction structurale pour les résidus appartenant à cette classe. Nous explorons enfin plusieurs pistes pour intégrer l'information des séquences homologues afin d'améliorer la prédiction par nos modèles.

Abstract

De novo methods for protein structure prediction aim at providing prediction for hard cases. Those methods usually require local structure prediction as a first step. The local structure of a protein can be described using the concepts of secondary structures such as helices, sheets and coils. We propose a method based on hidden Markov models to perform the local structure prediction. Hidden Markov models offer a strong theoretical background. Moreover, they allow an explicit data modeling.

We first develop a new method for protein secondary structure assignment. We investigate the use of hidden Markov models with only three hidden states and various memory schemes for protein secondary structure prediction. Then we increase the number of hidden states using two strategies : a model design based on previous knowledge about secondary structure, or the choice of the optimal model using statistical and accuracy criteria. As secondary structure prediction provides no clue about the structure of coil regions, we include geometrical descriptors (dihedral angle zones) of the coil in our models. We finally investigate several means to include the homologous sequence information to improve the prediction by hidden Markov models.