Projet Long :
Modélisation comparative par satisfaction de contraintes spatiales

Malo Leprohon
Encadrant : Jean-Christophe Gelly

M2BI
Année 2019-2020

Introduction :

La prédiction de la structure tridimensionnelle d'une protéine à partir de la séquence en acides aminés fait partie des enjeux majeurs en bio-informatique[1]. La détermination expérimentale de la structure tridimensionnelle d'une protéine est aujourd'hui réalisable dans de nombreux cas, notamment via les méthodes de cristallographie aux rayons X, de résonance magnétique nucléaire et de microscopie électronique. Cependant ces méthodes expérimentales peuvent être très coûteuses en temps et en argent. De nombreuses méthodes ont donc été développées à ce jour dans le but de prédire *in silico* la structure de protéine. Elle ne sont toutefois pas encore satisfaisantes dans de nombreux cas.

On peut regrouper les différentes méthodes de prédiction en trois grandes catégories. La première regroupe les méthodes *de novo*, qui cherchent à prédire la structure d'une protéine uniquement à partir de la séquence en acides aminés, en utilisant par exemple des méthodes de dynamique moléculaire ou de prédiction de structures secondaires. On retrouve parmi ces méthodes des logiciels comme QUARK[2] et ROSETTA[3]. La deuxième catégorie regroupe les méthodes par threading moléculaire. Ces méthodes cherchent à prédire la structure d'une protéine en s'appuyant sur des structures de protéines connues qui ont un repliement proche de la structure à prédire. Le choix de protéines supports est principalement basé sur nos connaissances sur la relation entre la structure d'une protéine et sa séquence. Des logiciels comme RAPTOR[4] et HHsearch[5] utilisent le threading moléculaire pour la prédiction de structures tridimensionnelles de protéines. La troisième catégorie de méthodes regroupe les méthodes de prédiction par homologie qui consistent à s'appuyer sur la structure connue d'une protéine homologue comme support. Ce type de méthode est actuellement le plus efficace mais requiert l'existence de structure homologue connue pour la structure à modéliser. La prédiction de structure par homologie repose principalement sur les étapes de choix d'une protéine homologue support et de prédiction de la structure de la protéine, à partir de la structure de la protéine support.

La méthode ORION (Optimized fold RecognitION)[6] est une méthode de reconnaissance de repliement qui permet l'identification de structures homologues lointaines ou partielles. Cette méthode a déjà été utilisée à deux reprises lors de la compétition internationale CASP (Critical Assessment of protein Structure Prediction)[7] en 2014 et 2018 où elle a réalisée de bonnes performances. La méthode ORION utilise ensuite une modélisation comparative basée sur les logiciels Modeller[8] et ROSETTA[9]. Ces deux logiciels font partie des plus performants en modélisation comparative mais présentent toutefois quelques limites et possèdent des licences d'exploitation restrictives.

L'objectif de ce projet long est de répondre à ce problème en développant un programme de modélisation comparative. Ce programme part d'un alignement d'une séquence cible à modéliser avec une séquence support dont la structure tridimensionnelle est connue, pour prédire la structure de la protéine cible à partir de la structure support. Le programme est principalement basé sur le logiciel de dynamique moléculaire GROMACS[10] pour des calculs d'énergie et l'application de contraintes spatiales. Ces contraintes sont issues de la structure support et sont des contraintes de distances entre les atomes de la chaîne principale (carbone, azote et carbone alpha) ou d'angles dièdres de la chaîne principale (angles phi, psi et oméga). GROMACS possède lui l'avantage d'être un logiciel libre (GNU Lesser Public, licence Version 2.1) et d'avoir de très bonnes performances en terme de temps de calcul. La première étape du projet était de réimplémenter et améliorer une préversion du programme en Python3. La seconde étape consistait à implémenter la prise en compte de la proximité d'une position avec des indels, dans son alignement avec la séquence support, pour l'intensité des contraintes qui lui sont imposées.

Matériels et méthodes :

Logiciels et packages utilisés :

Le programme est majoritairement développé en Python3 et utilise les modules suivants :

- Biopython (version 1.75) qui est utilisé pour la lecture de fichiers pdb, le calcul de distances et d'angles dièdres.
- Peptide Builder (version 1.0.4) qui est utilisé pour créer une structure linéaire de la protéine cible.

Les logiciels suivants sont utilisés à différentes étapes du programme :

- GROMACS (version 2016.4). Cette version précise de GROMACS est utilisée pour toutes les étapes de dynamique moléculaire.

- mkdssp (version 3.0) est utilisé pour obtenir l'information des structures secondaires de la protéine support.

Algorithme :

L'algorithme du programme est divisé en deux principales étapes (Figure 1). La première est une étape d'initialisation où les contraintes à appliquer sont calculées à partir de la structure support et où une structure linéaire de la protéine cible est créée. La seconde étape réalise le repliement de la protéine cible, à l'aide des contraintes précédemment générées par dynamique moléculaire en utilisant GROMACS.

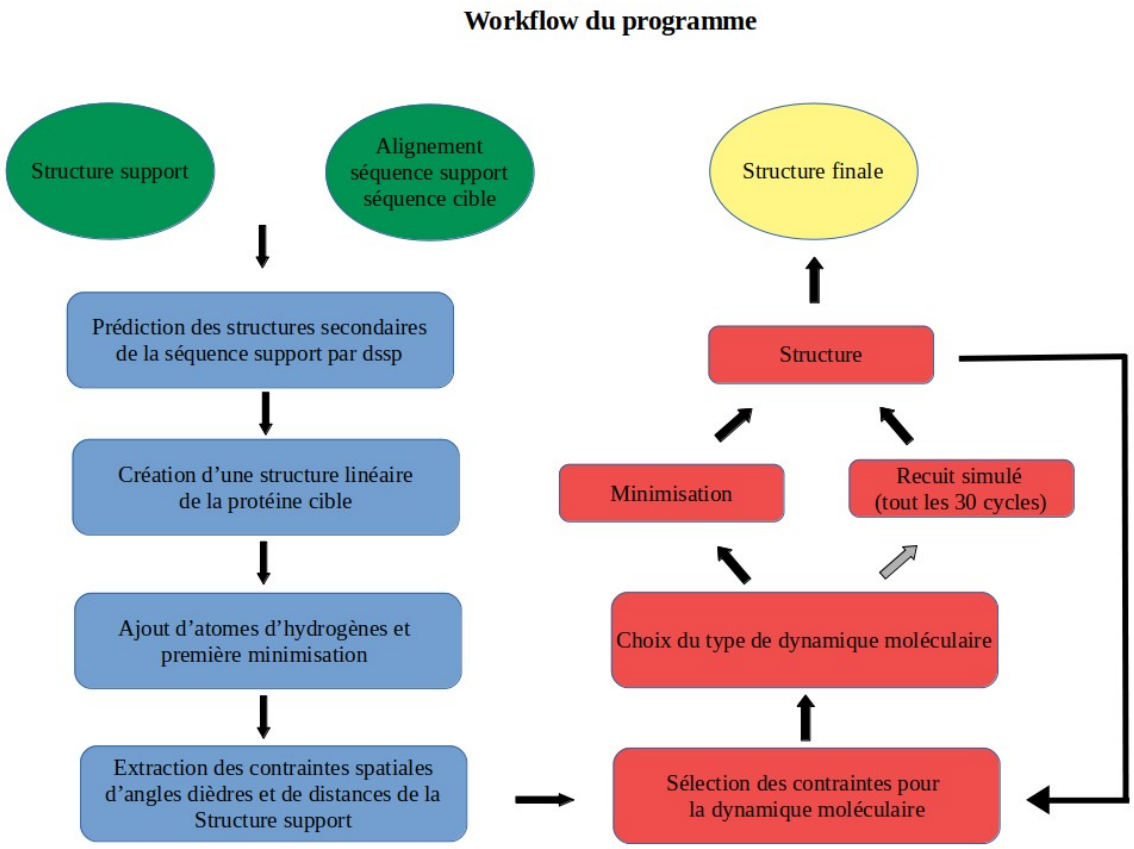


Figure 1 : Schéma des différentes étapes réalisées par le programme.

Initialisation de la structure de la protéine cible :

Le but de cette étape est d'initialiser une structure linéaire de la protéine linéaire. Le programme va en premier lieu réaliser un dssp sur la structure de la protéine support pour obtenir l'information sur la structure secondaire de la protéine support. Une structure linéaire de la protéine est ensuite construite en assignant des valeurs d'angle phi et psi qui se rapprochent de celle d'un brin bêta (-120, 120). Une correspondance entre les positions de la séquence cible et de la séquence support est ensuite réalisée. Ces correspondances sont alors utilisées pour assigner des valeurs d'angles phi et psi d'hélices alpha (-57,8 , -47) aux positions de la séquence cible qui correspondent à des positions de la séquence support qui appartiennent à des hélices alpha selon les données issues du dssp. Peptide Builder est alors utilisé pour construire une séquence linéaire de la protéine dans un fichier pdb à partir des valeurs d'angles phi et psi précédemment assignées.

GROMACS est ensuite utilisé pour ajouter des atomes d'hydrogène à cette première structure. Puis celle-ci subit une première étape de minimisation pour préparer les fichiers nécessaires à l'étape de repliement.

Calcul des contraintes spatiales à partir de la structure de la séquence support :

Les contraintes spatiales de distance et d'angle dièdre sont calculées à partir de la structure de la protéine support. Les valeurs de distance entre les atomes de la chaîne principale (carbone, azote, carbone alpha) sont calculées et extraites de la structure support ainsi que les valeurs des angles phi, psi et oméga. Le programme utilise ensuite la correspondance entre les positions de la séquence cible et les positions de la séquence support pour assigner ces valeurs de distances et d'angles dièdres à des positions de la séquence cible. Cela permet également d'éliminer les valeurs qui n'ont pas de correspondance entre la position de la séquence support et la position de la séquence cible. Le programme va aussi corriger les valeurs de contraintes spatiales pour les résidus proches de positions indels dans l'alignement. Les positions distantes de 1, 2 ou 3 positions d'un indel auront des contraintes spatiales atténuées de 100 %, 66 % et 33 % respectivement. Enfin les contraintes d'angles dièdres et de distances sont respectivement écrites au format GROMACS (voir la documentation de GROMACS) dans deux fichiers différents.

Repliement progressif de la structure linéaire de la protéine cible par dynamique moléculaire et applications des contraintes spatiales :

Une fois la structure linéaire de la protéine cible construite et les contraintes spatiales extraites de la structure de la protéine support , la structure de la protéine cible est repliée progressivement par un certains nombres d'étapes de dynamique moléculaire. Ce repliement est effectué à l'aide des contraintes spatiales. L'ensemble des contraintes d'angles dièdres sont appliquées tout au long du repliement. Cependant les contraintes de distances sont appliquées progressivement et de manière itérative tout au long du repliement. C'est à dire qu'une position de la structure de la protéine cible ne subira initialement que les contraintes de distance au niveau local, avec les atomes des positions voisines. Au fur et à mesure du repliement, la position subira des contraintes de distances supplémentaires avec des positions de plus en plus éloignées, jusqu'à englober l'ensemble des positions de la protéine. Cela permet un repliement au niveau local, puis de plus en plus global de la structure pour aboutir à un modèle de la structure de la protéine cible. Le nombre d'étapes de dynamique moléculaire effectuées est légèrement supérieur à la longueur de la chaîne peptidique afin de prendre en compte l'ensemble des contraintes de distances à la fin du repliement. La structure générée à l'issue de l'ensemble des étapes de repliements représente le modèle de structure tridimensionnelle de la protéine cible prédit par le programme.

Déroulement d'une étape de dynamique moléculaire :

À chaque étape de dynamique moléculaire, un fichier de contraintes spatiales de GROMACS (fichier.itp) contenant les contraintes de distances sélectionnées et les contraintes d'angles dièdres est écrit. Ensuite un fichier de dynamique moléculaire (fichier.mdp) est choisi pour l'étape de dynamique moléculaire. Le programme peut choisir un fichier de minimisation qui entraînera une dynamique moléculaire de minimisation. Ce type de dynamique concerne les étapes de repliements classiques où les contraintes sont appliquées, GROMACS essaye alors de minimiser l'énergie potentielle de la molécule en prenant en compte ces contraintes. Cependant toutes les trente étapes de dynamique moléculaire, le programme choisira un second fichier de dynamique moléculaire qui lancera une dynamique moléculaire en recuit simulé. Ce type d'énergie permet d'explorer différentes conformations de la protéine et de sortir de minima locaux qui pourraient stopper la protéine dans son repliement, malgré la présence de contraintes spatiales. Une fois les contraintes spatiales et le type de dynamique déterminés, la dynamique moléculaire est effectuée par GROMACS.

Paramètre de dynamique moléculaire :

Minimisation avec GROMACS:

La minimisation effectuée par GROMACS est une minimisation de 150 étapes de 0,001 ps pour une dynamique de 0,150 ps. La minimisation est effectué par gradient-conjugué. L'algorithme d'interaction électrostatique est le PME (Particle Mesh Ewald).

Recuit simulé avec GROMACS :

Le recuit simulé effectué par GROMACS est un recuit simulé de 5000 étapes de 0,002 ps pour une dynamique de 10 ps. Le couplage de température se fait par l'algorithme de Berendsen et le couplage de la pression se fait par l'algorithme de Parrinello-Rhaman. L'annealing se fait en trois points à 0, 5 et 10 ps à des température de 0, 300 et 500 Kelvin respectivement.

Prédiction de structure tridimensionnelle :

Différentes prédictions de structure tridimensionnelle ont été réalisées pour tester le fonctionnement du programme. Les différents peptides utilisés ont été choisis depuis la base de données d'alignement de famille de protéine homologue HOMSTRAD (HOMologous STRucture Alignment Database).

La structure de la chaîne A de l'insuline hunaine (code PDB : 3i40) a été prédite à partir de sa propre structure. Le but de ce test était de vérifier le fonctionnement du programme sur un cas très simple et rapide à calculer.

La structure de la chaîne A de la léghémoglobine de soja (code PDB : 1fsl) a été prédite à partir de sa propre structure. Le but de ce test était de vérifier le fonctionnement du programme sur un cas un peu plus complexe et de vérifier la bonne stabilité des étapes de dynamique moléculaire.

La structure du domaine d'homologie SRC 3 (SH3) (code PDB : 1shg) a été prédite à partir de la structure du complexe intramoléculaire ITK-Proline (code PDB : 1awj). Le but de ce test était de

vérifier le fonctionnement du programme sur un cas réel, ainsi que l'influence des indels sur les différentes contraintes.

La prédiction inverse de celle précédemment décrite a également été réalisée avec le même objectif que la prédiction précédente.

Les différents modèles prédits ont été alignés, en omettant les chaînes latérales, à leur structure dans la PDB (Protein Data Bank). Le RMSD (Root Mean Square Deviation) des positions atomiques a été calculé pour évaluer la qualité des modèles.

Disponibilité et test :

Le programme est disponible sur le lien git hub suivant :
https://github.com/leprohonmalo/Projet_Long

Le git hub contient l'ensemble des scripts du programme dans le dossier source et les fichiers pir et pdb utilisés pour tester le programme dans le répertoire data.

Pour lancer le programme il faut exécuter le script toto.py situé dans le dossier src. Le script prend en argument un fichier pdb représentant la structure de la protéine support, un fichier de pir contenant la séquence support alignée à la séquence cible, un répertoire existant où seront écrits l'ensemble des fichiers de sorties. Le script peut également prendre deux valeurs en temps qu'arguments optionnels représentant la force des contraintes spatiales de distances et d'angles dièdres qui seront appliquées à la structure de la protéine cible.

Le fichier pdb doit contenir uniquement des atomes issus de la chaîne peptidique cible. La première séquence alignée dans le fichier pir est considérée par le programme comme la séquence support. La seconde séquence contenue dans le fichier pir est considérée par le programme comme la séquence cible.

Exemple de ligne de commande permettant de lancer le programme sur Linux :
./toto.py exemple/3i40_A.pdb exemple/3i40.pir exemple/results/3i40 --dist_cons 200 --angles_cons 50

Le programme génère de nombreux fichiers, parmi eux on retrouve :

- des fichiers minN.pdb. Ces fichiers contiennent la structure de la protéine cible après la N-ième étape de dynamique moléculaire. Le dernier fichier minN.pdb représente la structure de la protéine cible prédite par le programme.
- des fichiers minN.itp qui contiennent les contraintes spatiales appliquées lors de l'étape de dynamique moléculaire N.
- d'autres fichiers générés par GROMACS (.gro, .top, .tpr, .trr, .edr, .mdp, .log). Plus d'informations à leurs propos sont disponibles dans la documentation de GROMACS.
- un fichier template.dssp qui correspond au fichier créé à partir des prédictions de structures secondaires de la structure de la protéine cible par dssp.
- des fichiers first_structure qui sont les fichiers à propos de la structure linéaire initiale de la protéine cible.
- des fichiers .tmp qui sont des fichiers temporaires utilisés pour le stockage de données et qui seront supprimés après exécution dans une version future du programme.

Résultats :

Prédiction de la structure de la chaîne A de l'insuline humaine :

Pour tester le programme dans un premier temps, la structure de la chaîne A de l'insuline (3i40) a été prédite par le programme en utilisant sa propre structure comme protéine support. Ce peptide est très petit (21 acides aminés) et comporte une structure très simple, très facile à modéliser. La figure 2 permet de visualiser différentes structures adoptées par cette chaîne peptidique au cours de différentes étapes de repliement. On observe un repliement progressif du peptide, qui part d'une structure linéaire possédant deux hélices alpha pour arriver à une structure possédant deux hélices alpha un peu plus importantes, agencées en épingles. La structure prédite de la chaîne A de l'insuline a ensuite été évaluée par alignement avec la chaîne principale de la structure de référence utilisée comme protéine support (Figure 3). On observe que la structure prédite et la structure de référence de la chaîne A de l'insuline sont très bien alignées et semblent très similaires au niveau de la chaîne principale. Cela est confirmé par une valeur de RMSD très faible égale à 1,773 Angströms. L'alignement est de bonne qualité, aussi bien au niveau des structures secondaires qu'au niveau de régions de random coil. Les régions de random coil sont celles où on retrouve le plus de dissimilarité dans l'alignement. Cette observation est attendue puisque ces régions sont souvent les plus flexibles en dynamique moléculaire et *in vivo*. La qualité de l'alignement montre que les contraintes appliquées lors des étapes de dynamique moléculaire ont été efficaces.

Cependant si on s'intéresse aux chaînes latérales on remarque que les conformations de celles-ci sont souvent très différentes entre la structure prédite et la structure de référence. Cela est plutôt attendu étant donné qu'aucune contrainte spatiale n'est appliquée sur les chaînes latérales des résidus de la protéine cible.

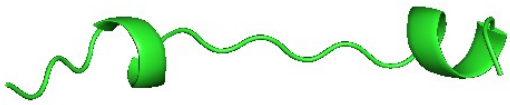


Figure 2a :Première structure linéaire de 3i40.

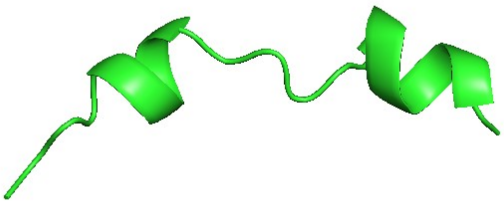


Figure 2b : Structure de 3i40 à la dynamique 6

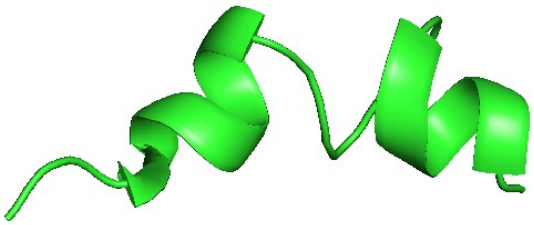


Figure 2c : Structure de 3i40 à la dynamique 12

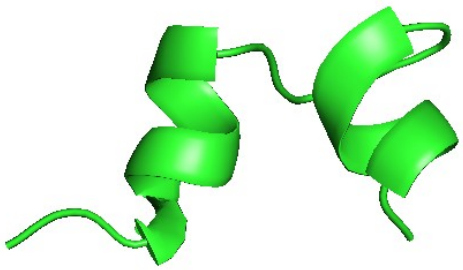


Figure 2d: Structure de 3i40 à la dynamique 18

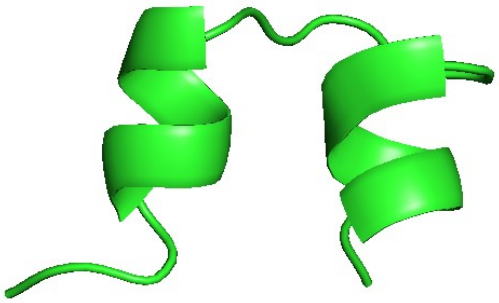


Figure 2e: Structure de 3i40 à la dynamique 24

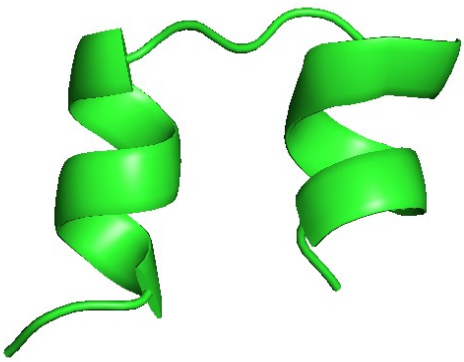


Figure 2f: Structure de 3i40 à la dynamique finale (27)

Figure 3: Alignement de la structure de la chaine A de 3i40 (cyan) avec la structure prédite par le programme de la chaîne A de 3i40 (vert). Le RMSD est égal à 1,773 A pour un alignement des carbone alpha.

Prédiction de la structure de la chaîne A de la léghémoglobine de soja :

Dans un deuxième temps le programme a été testé sur une structure plus complexe, la chaîne A de la léghémoglobine de soja (1bina). Cette chaîne peptidique de 143 acides aminés possédant une structure avec plusieurs couches d'hélices alpha dans différentes orientations. La Figure 4 permet de visualiser différentes structures adoptées par cette chaîne peptidique au cours de différentes étapes de repliement. On observe un repliement progressif du peptide, qui part d'une structure linéaire possédant sept hélices alpha pour arriver à une structure possédant sept hélices alpha un peu plus importantes, agencées en protéine globulaire.

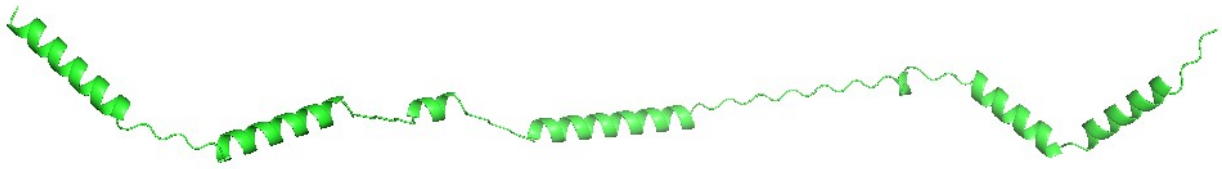


Figure 4a: Première structure linéaire de 1bina.

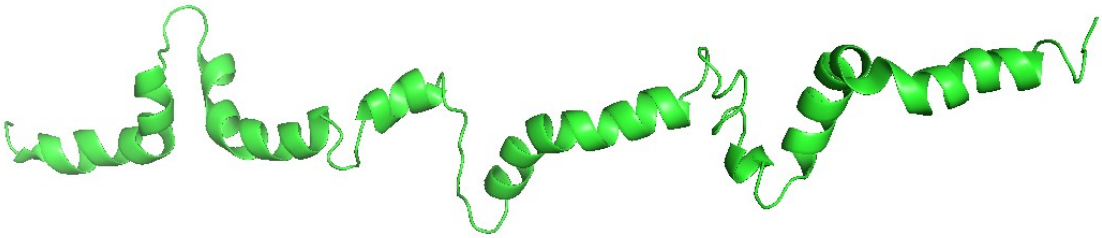
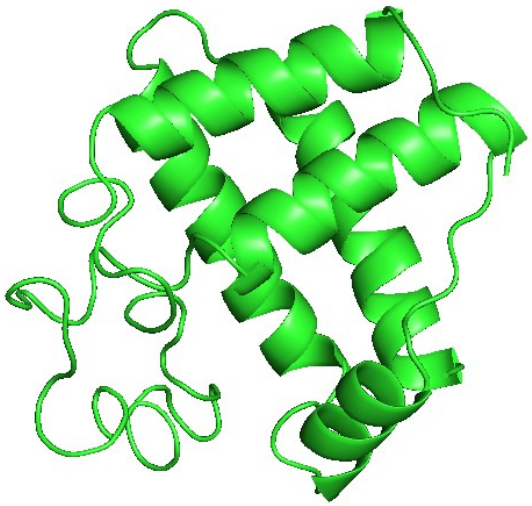


Figure 4b: Structure de 1bina à la dynamique 30

Figure 4c: Structure de 1bina à la dynamique 60



IFigure 4d: Structure de 1bina à la dynamique 90

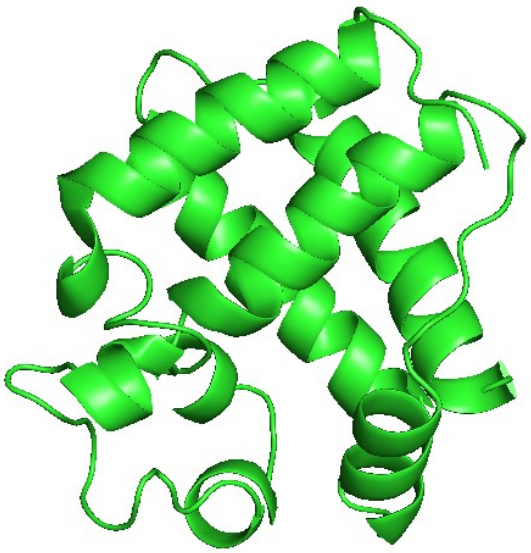


Figure 4e: Structure de 1bina à la dynamique 120

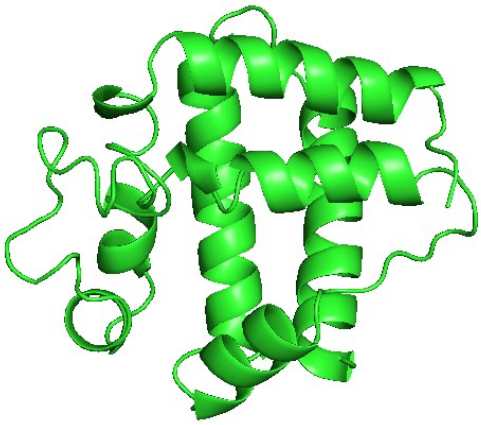


Figure 4f: Structure de 1bina à la dynamique finale (149)

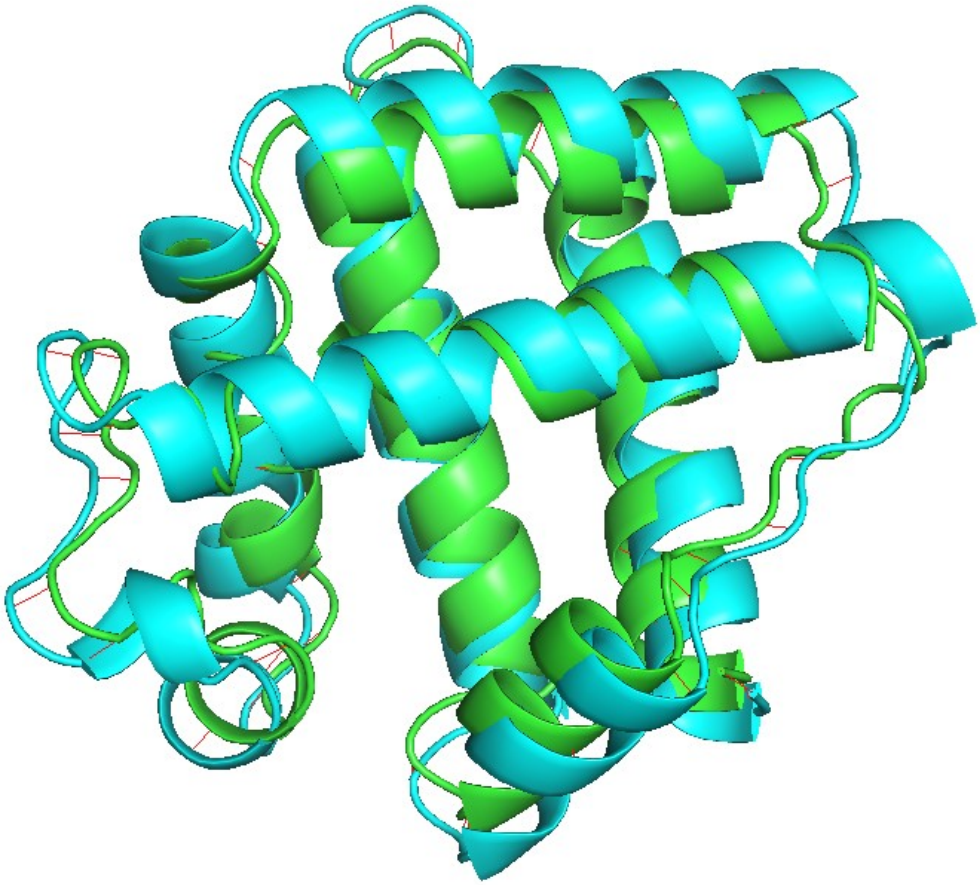


Figure 5: Alignement de la structure de 1bina (cyan) avec la structure prédite par le programme de 1bina (vert). Le RMSD est égal à 1,471 A pour un alignement des carbone alpha.

L'alignement de la structure prédite avec la structure de référence de la PDB (Figure 5) montre que la structure prédite est très proche de la structure de référence, le RMSD égal à 1,471 Angströms est même plus faible que dans le modèle de la chaîne A de l'insuline. Cela montre encore une fois que les contraintes spatiales ont été efficaces lors des étapes de dynamique moléculaire. La prédiction de cette structure de taille plus importante à aussi permis de tester la stabilité des dynamiques moléculaires de minimisation et de recuit simulé. Si les étapes de minimisation se sont montrées très stables, ce n'est pas le cas des étapes de recuit simulé qui rencontrent trop souvent une erreur pour pouvoir être appliquées par le programme. Le problème n'ayant pas encore pu être corrigé, les prédictions de structures tridimensionnelles ont été réalisées sans étapes de recuit simulé.

Prédictions des structures de SH3 et du complexe ITK-Proline :

Pour tester la capacité du programme à gérer les contraintes spatiales autour des indels, la structure de SH3 a été prédite à partir de la structure du complexe ITK-Proline et inversement. Cependant , aucun repliement des protéines cibles n'a pu être observé dans les structures produites par le

programme (résultat non montré). Le problème n'a pas encore pu être corrigé et il est nécessaire de comprendre son origine pour corriger le programme.

Discussion :

Les tests effectués sur le programme permettent de constater son bon fonctionnement dans le cas de la prédiction de la structure tridimensionnelle d'une protéine de petite taille. Cependant le programme est toujours en cours de développement et son état actuel ne permet pas encore la prédiction de structure par homologie. En effet les étapes de dynamiques moléculaires ne semblent pas encore au point et il reste à déterminer l'origine des problèmes qui empêchent leur fonctionnement. Ces problèmes pourraient venir de la paramétrisation de la dynamique moléculaire, d'une trop faible correspondance entre les séquences (mais des contraintes sont appliqués sur plus de 50 % de la protéine donc cela est peu probable) ou une erreur d'implémentation. L'absence de recuit simulé pourrait aussi participer à empêcher le repliement, et parvenir à implémenter ce type de dynamique dans le programme pourrait améliorer les prédictions.

Le programme pourrait aussi être amélioré par l'implémentation d'autres fonctionnalités pour améliorer les prédictions. Une étape de choix de dimensions de la boîte de dynamiques moléculaires, adaptées à la protéine cible serait intéressante. Cela pourrait prévenir la rencontre de certains problèmes de dynamiques moléculaires et de conflits entre les dimensions de la boîte et les contraintes spatiales de distances. Il serait également intéressant d'accorder plus de liberté à l'utilisateur sur certains aspects du programme, comme le nombre de d'étapes de repliement, la fréquence des étapes de recuit simulé, l'intensité et la portée de l'atténuation des contraintes spatiales pour des positions proches d'indels, ou encore l'intégration des propres contraintes spatiales de l'utilisateur dans le repliement de la structure linéaire de la protéine cible.

De nombreux autres tests de prédictions avec des structures variées pourraient également permettre d'optimiser certains aspects du programme comme l'atténuation des contraintes spatiales à proximité des indels ou les paramètres de dynamiques moléculaires. Une fois le programme réellement capable de réaliser une modélisation comparative, il sera aussi intéressant de comparer les performances de prédictions du programme avec les performances d'autres logiciels de modélisation comparative comme Modeller. Si ces logiciels ont de meilleurs performances sur des cas spécifiques, on pourrait essayer de comprendre pourquoi afin d'améliorer le programme.

Bibliographie :

1. Kennedy, D. What Don't We Know? *Science* **309**, 75–75 (2005).
2. Xu, D. & Zhang, Y. Ab initio protein structure assembly using continuous structure fragments and optimized knowledge-based force field. *Proteins Struct. Funct. Bioinforma.* n/a-n/a (2012) doi:10.1002/prot.24065.
3. Raman, S. *et al.* Structure prediction for CASP8 with all-atom refinement using Rosetta. *Proteins Struct. Funct. Bioinforma.* **77**, 89–99 (2009).
4. Källberg, M. *et al.* Template-based protein structure modeling using the RaptorX web server. *Nat. Protoc.* **7**, 1511–1522 (2012).

5. Fidler, D. R. *et al.* Using HHsearch to tackle proteins of unknown function: A pilot study with PH domains. *Traffic* **17**, 1214–1226 (2016).
6. Ghouzam, Y., Postic, G., Guerin, P.-E., de Brevern, A. G. & Gelly, J.-C. ORION: a web server for protein fold recognition and structure prediction using evolutionary hybrid profiles. *Sci. Rep.* **6**, 28268 (2016).
7. Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T. & Tramontano, A. Critical assessment of methods of protein structure prediction (CASP)-Round XII. *Proteins Struct. Funct. Bioinforma.* **86**, 7–15 (2018).
8. Eswar, N. *et al.* Comparative Protein Structure Modeling Using Modeller. *Curr. Protoc. Bioinforma.* **15**, 5.6.1-5.6.30 (2006).
9. Song, Y. *et al.* High-Resolution Comparative Modeling with RosettaCM. *Structure* **21**, 1735–1742 (2013).
10. Berendsen, H. J. C., van der Spoel, D. & van Drunen, R. GROMACS: A message-passing parallel molecular dynamics implementation. *Comput. Phys. Commun.* **91**, 43–56 (1995).