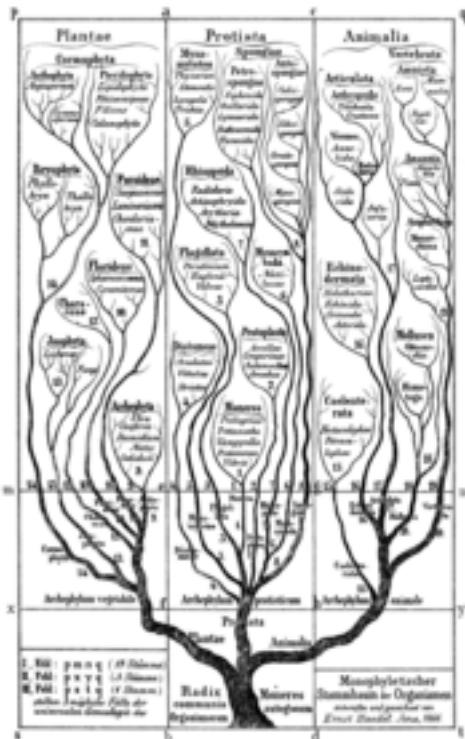


Gabriel Chandesris - Février 2007

Projet Tutoré de Licence Professionnelle
BioTechnologie option Bio-Informatique

Phylogenetik

**Conception, développement et tests
d'un logiciel en Java
pour la construction d'arbres phylogénétiques
avec une interface graphique.**



Arbre de classification des espèces, Haeckel, 1866

• Résumé - Introduction

La phylogénie est l'étude de la formation des êtres vivants et de leur parenté. Cette parenté est représentée par des arbres phylogénétiques dont les branches indiquent le degré de parenté. Plusieurs méthodologies existent pour construire ces arbres :

- l'approche cladistique, où le degré de parenté est calculé et hiérarchisé selon l'homologie de caractères entre les organismes, le regroupement est alors fait si l'homologie est estimée significative (plusieurs organismes partageant un même caractère),
- l'approche phénétique, où le degré de parenté est corrélé au degré de ressemblance, notamment expliqué par la convergence évolutive, cette approche est pertinente quand un grand nombre de critères est comparé entre organismes.

Cette dernière approche s'effectue notamment par un grand nombre de comparaisons, notamment au niveau moléculaire, par la comparaison et l'alignement de groupes de séquences protéiques et l'alignement de groupes de séquences nucléiques, ces séquences appartenant à plusieurs organismes.

L'alignement des séquences permet de calculer la similarité entre séquences, et ainsi de construire un arbre phylogénétique de ces séquences. Par déduction, on peut ainsi construire un arbre de classification des espèces d'où proviennent ces séquences.

Le grand nombre de séquences à comparer, ainsi que la répétition des calculs, font de ce travail de construction d'arbres phylogénétiques un travail idéal pour une machine à calculer qu'est un ordinateur, au moins pour les alignements de séquences et leur regroupement pour construire les arbres phylogénétiques. De nombreux programmes et logiciels existent déjà pour réaliser ce travail, de qualité inégale, autant pour leur aspect graphique que pour leur fonctionnement. Leur mode de diffusion est libre ou commercial.

L'objectif de ce projet est de fournir un logiciel de construction d'arbres en Java utilisant les algorithmes UPGMA et Neighbour-Joining : Phylogenetic. Ce logiciel doit comporter une interface facile à utiliser. L'affichage des arbres doit correspondre à des structures disponibles dans les bibliothèques Java.

Le fonctionnement du logiciel obtenu est explicité par des exemples connus de classification (les différentes souches du VIH). Le développement du logiciel Phylogenetic est limité, différentes possibilités d'évolution de ce programme sont présentées.

Table des matières

.	RÉSUMÉ - INTRODUCTION	2
.	TABLE DES MATIÈRES.....	3
.	ABRÉVIATIONS	3
.	TABLE DES ILLUSTRATIONS	3
.	RÉALISATION DU PROJET	4
1.	STRUCTURE DU PROGRAMME	4
2.	ALGORITHMES MIS EN PLACE	5
3.	INTERFACE GRAPHIQUE	6
3.1.	<i>Fenêtre principale</i>	6
3.2.	<i>Gestion des évènements.....</i>	7
3.3.	<i>Construction et affichage des arbres.....</i>	7
.	TESTS DE FONCTIONNEMENT	7
1.	EXEMPLES DE FONCTIONNEMENT.....	7
2.	RÉSULTAT DES TESTS.....	8
.	CONCLUSION ET AMÉLIORATIONS POSSIBLES	11
.	BIBLIOGRAPHIE ET RESSOURCES	11
.	ANNEXES	12

Abréviations

HIV	Human Imunodeficiency Virus – Virus de l’Immunodéficience Humaine
NJ	Neighbour-Joining
SIV	Simian Imunodeficiency Virus – Virus de l’Immunodéficience Simienne
UPGMA	Unweighted Pair Group Method with Arithmetic mean

Table des illustrations

FIGURE 1 : DIAGRAMME DE FONCTIONNEMENT DU LOGICIEL PHYLOGENETIK	5
FIGURE 2 : FENETRE PRINCIPALE DU LOGICIEL PHYLOGENETIK.	6
FIGURES 3 ET 4 : ARBRES PHYLOGÉNÉTIQUES POUR LA PROTÉINE ENV (UPGMA ET NJ).	8
FIGURES 5 ET 6 : ARBRES PHYLOGÉNÉTIQUES POUR LA PROTÉINE TAT (UPGMA ET NJ).....	8
FIGURES 7 ET 8 : ARBRES PHYLOGÉNÉTIQUES POUR LA PROTÉINE ENV RÉSIDUS 1 À 100.....	9
FIGURES 9 ET 10 : ARBRES PHYLOGÉNÉTIQUES POUR LA PROTÉINE ENV RÉSIDUS 101 À 200.....	9
FIGURES 11 ET 12 : ARBRES PHYLOGÉNÉTIQUES POUR LA PROTÉINE ENV RÉSIDUS 201 À 300.....	10
FIGURES 13 ET 14 : ARBRES PHYLOGÉNÉTIQUES POUR LA PROTÉINE ENV RÉSIDUS 301 À 400.....	10

. Réalisation du projet

Le projet a été réalisé en Java 1.4.2 en utilisant la bibliothèque graphique Swing, avec Eclipse 3.0. Cette utilisation sera détaillée par la suite dans la partie correspondant à l'interface graphique, pour les fenêtres, panneaux, boutons, zones de textes et gestions d'évènements.

1. Structure du programme

La classe *Phylogenetik* comporte tous les éléments de l'interface graphique et les éléments afférents à son affichage et à la gestion des événements liés à cette interface : ajout de séquence, demande de sélection de fichier, affichage du nom et de la séquence sélectionnée, lancement de la construction d'arbres phylogénétiques.

Une instance de cette classe permet le lien entre l'utilisateur et les calculs effectués par l'application (capture d'événements, transmission et affichage des données), et aussi la possibilité d'effectuer les tâches de l'application de façon transparente.

PhyloTools est la classe qui comporte toutes les méthodes utiles à la construction des arbres phylogénétiques, c'est-à-dire l'enregistrement des séquences, l'alignement des séquences deux à deux, l'enregistrement des scores d'alignement dans un tableau d'entiers (matrice de score) et les algorithmes correspondant à la construction d'arbres.

Une seule instance de cette classe fait le lien entre la liste des séquences introduites pour le calcul de l'arbre, la matrice de scores obtenue avec les classes d'alignements et les méthodes de construction des arbres (UPGMA et Neighbour-Joining). L'enregistrement des séquences est effectué par le regroupement dans un tableau d'instances de la classe *LifeSequence*.

Les alignements sont effectués par l'utilisation des classes suivantes : *Alignment*, *AlignNW*, *AlignSW* et *MatrixSub*. Les trois premières classes servent à l'alignement (comparaison de deux séquences) en utilisant une matrice de substitution spécifique - *transition / transversion* pour les séquences d'acides nucléiques ou *blosum50* pour les séquences protéiques.

Les classes d'alignements ne sont instanciées que pour le calcul et l'obtention des scores d'alignement des séquences deux à deux. Ces classes reconnaissent automatiquement le type des séquences (protéique ou nucléique) et utilisent des matrices de substitution en conséquence.

Les séquences sont mises en mémoire par le logiciel dans une structure de données particulière, *LifeSequence*, qui comporte le nom et la séquence, ainsi que des méthodes spécifiques à ces données pour déterminer le type des séquences. Le logiciel est prévu pour fonctionner avec des fichiers sources de séquences au format FASTA.

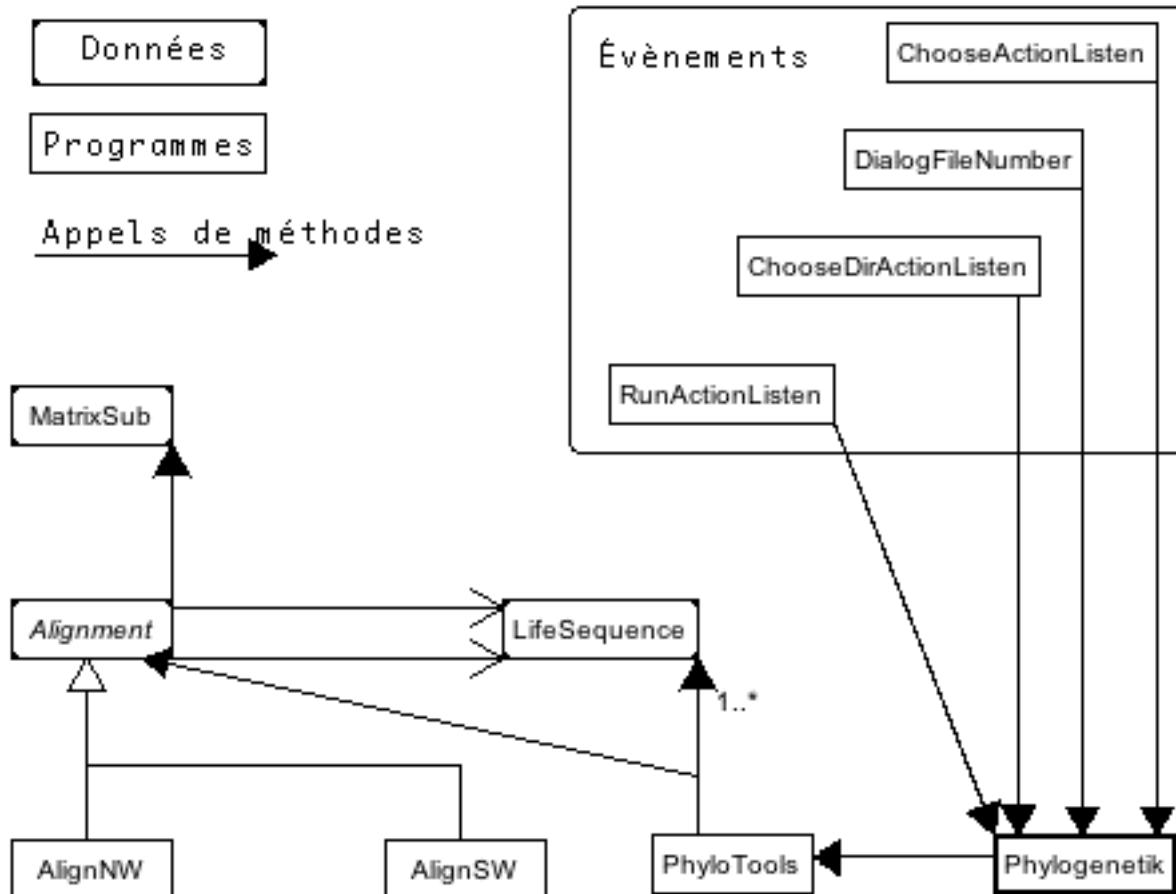


Figure 1 : Diagramme de fonctionnement du logiciel Phylogenetic

2. Algorithmes mis en place

Les méthodes d'alignements ont été regroupées en classes spécifiques, du fait du traitement des données et pour faciliter leur utilisation et réutilisation dans d'autres projets. C'est l'alignement de Needelman et Wunsch qui est utilisé par défaut pour l'alignement et le calcul des scores entre deux séquences, l'algorithme de Smith et Waterman est implémenté mais n'est pas utilisé.

Ces deux algorithmes sont mis en place dans deux classes distinctes héritant d'une classe générique d'alignement, toujours dans un souci de portabilité et d'amélioration possible du logiciel.

Les algorithmes de constructions d'arbres UPGMA et Neighbour-Joining ont été implémentés dans le programme sous forme de deux méthodes internes à la classe *PhyloTools*. Ces deux méthodes ont été conçues selon le même schéma : recopie de la matrice de score obtenue par les alignements, modifications de la matrice recopiée et utilisation pour construire l'arbre par regroupement des séquences dans des nœuds, initialisation du nœud racine de l'arbre obtenu.

Les résultats de ces algorithmes d'alignement et de construction d'arbres sont traités par la classe *PhyloTools* de façon à pouvoir ensuite être appelés par la classe *Phylogenetic* et affichés dans l'interface graphique sous forme de texte (matrices de scores) ou sous forme adaptée (affichage des arbres).

3. Interface graphique

Le logiciel Phylogenetik est une application en Java, utilisant la bibliothèque graphique Swing qui permet notamment de construire une interface graphique agréable en utilisant différents éléments d'affichage programmables. Un écran du logiciel est composé d'une fenêtre englobante (*JFrame*), composée de deux panneaux (*JPanel*), chacun de ces panneaux étant dédié à des fonctionnalités précises.

3.1. Fenêtre principale

Le premier panneau situé à gauche comporte des titres graphiques, une série de boutons cliquables (*JButton*) pour la sélection des séquences, et une zone de texte (*JTextArea*) pour l'affichage des noms des séquences au fur et à mesure des ajouts effectués par l'utilisateur.

Le deuxième panneau comporte un seul bouton (*JButton*) et une zone de texte (*JTextArea*), ces deux éléments servent respectivement à lancer les différents éléments du programme (alignement et construction des arbres) et à afficher les données des différentes étapes du fonctionnement (ajout de séquence, matrice de score et utilisation de cette matrice pour construire les arbres).

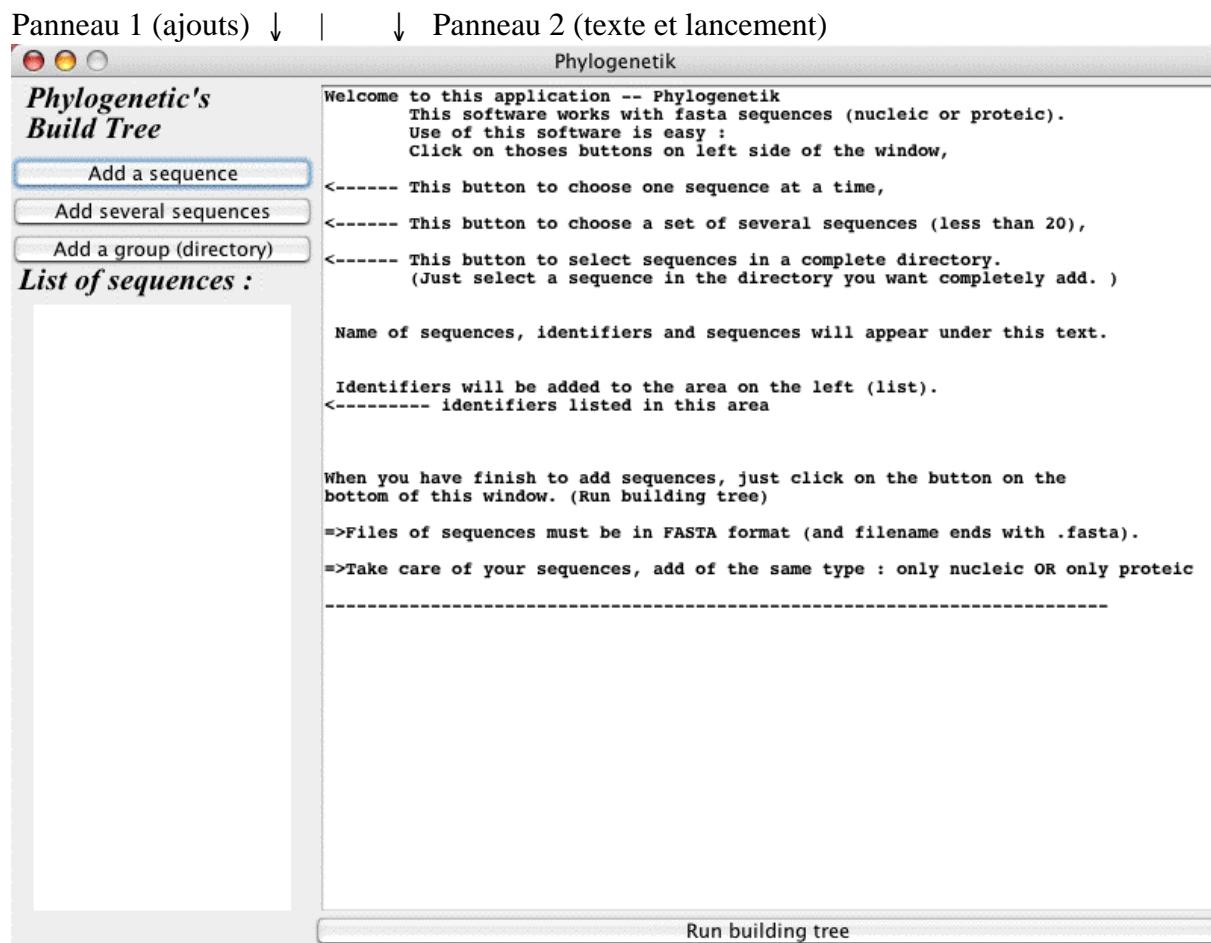


FIGURE 2 : FENETRE PRINCIPALE DU LOGICIEL PHYLOGENETIK.

3.2. Gestion des événements

L'interface graphique en Java comporte une gestion des évènements, notamment par l'utilisation des différents boutons de l'interface : ceux-ci implémentent le déclenchement d'événements, ces événements sont ensuite récupérés par des classes et des instances adaptées.

Pour lancer le choix d'un ou de plusieurs fichiers contenant les séquences, c'est l'événement *ChooseActionListen* qui est généré et utilisé pour demander la sélection d'une ou de plusieurs séquences. Dans ce dernier cas, une étape intermédiaire demande le nombre de séquences à sélectionner, *DialogFileName*.

Pour la sélection d'un ensemble de séquences contenues dans un répertoire un événement différent, *ChooseDirActionListen*, est utilisé pour demander la sélection d'une séquence dans le répertoire désiré ; le logiciel s'occupant du reste de la sélection.

Une fois la sélection des séquences terminée, la construction de l'arbre proprement dite n'est lancée qu'à partir de l'événement *RunActionListen*, lorsque le bouton correspondant a été cliqué par l'utilisateur. Cet événement lance successivement les méthodes internes aux classes du programme qui correspondent à l'alignement des séquences et à la construction de la matrice de score, puis à la création des arbres à partir de cette matrice de score.

3.3. Construction et affichage des arbres

Les algorithmes utilisés regroupent les séquences et groupes de séquences par paires, en fonction du score minimum trouvé dans la matrice (forte similarité), jusqu'à ne plus obtenir qu'un seul nœud racine.

Au sein de l'interface graphique Java-Swing, ce sont encore des fenêtres (*JFrame*) et des panneaux (*JPanel*), qui sont utilisés avec les structures des données Java de construction et d'affichage des arbres. Pour ces derniers, il s'agit de *DefaultMutableTreeNode* pour la gestion des nœuds, et de *JTree* pour l'affichage.

• Tests de fonctionnement

Au cours du développement, les tests de fonctionnement sur le logiciel et ses différentes composantes ont été effectués avec des séquences construites artificiellement de façon informatique afin de tester la robustesse du programme et que les algorithmes soient corrects. Les séquences artificielles au format texte ont été construites sur plusieurs critères : la longueur (séquences courtes et séquences longues) et sur le nombre à faire traiter par le logiciel pour construire les arbres.

Ces tests ont permis de valider les différentes étapes du fonctionnement du logiciel, l'alignement entre paires de séquences, la construction de la matrice de scores utilisée pour la construction des arbres, les deux algorithmes de construction d'arbre (UPGMA et Neighbour-Joining). D'autres tests ont été réalisés pour la conception de l'interface graphique et l'affichage des arbres en Java.

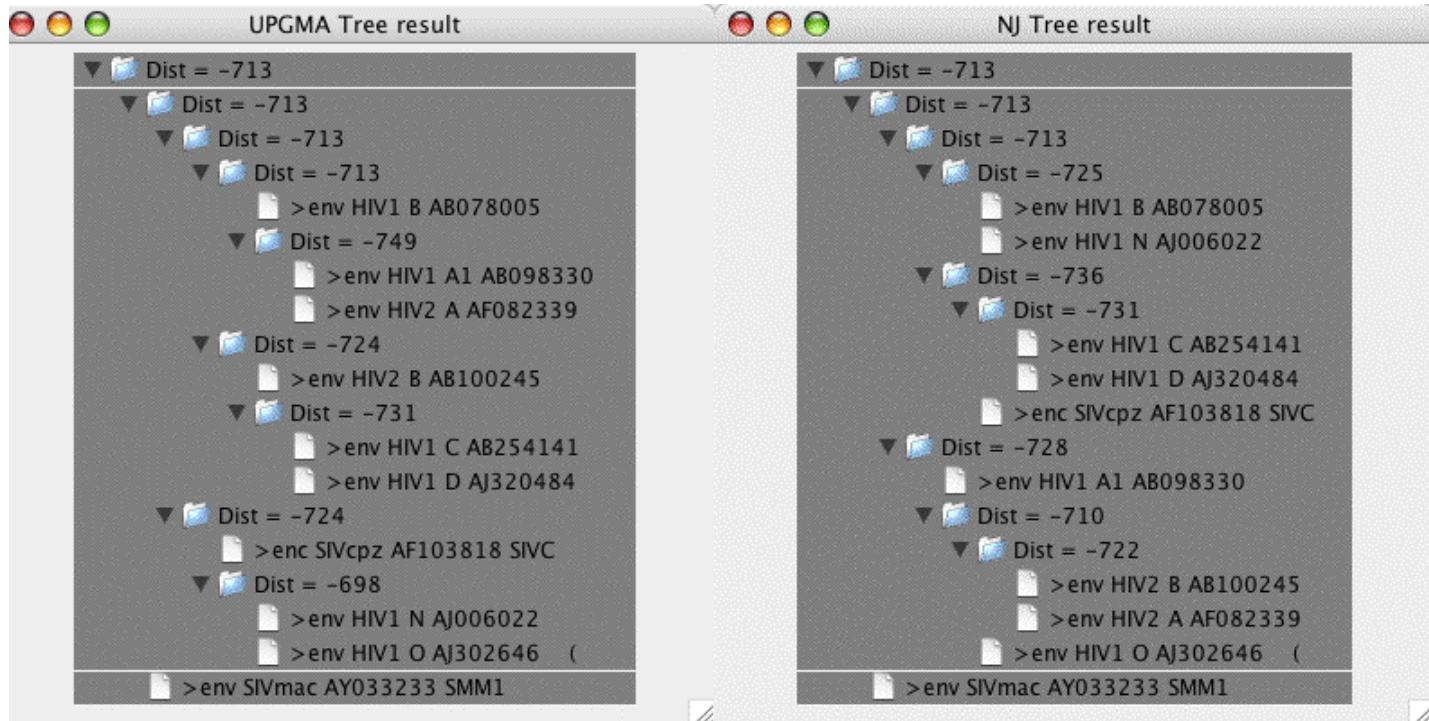
1. Exemples de fonctionnement

Le logiciel Phylogenetik a été mis en œuvre pour différents exemples de séquences réelles de protéines et d'acides nucléiques de différentes souches du virus du SIDA chez l'homme et le singe. Notamment les souches suivantes : VIH-1 (A, B, C, D, N et O), VIH-2 (A et B) ainsi que deux SIV, SIVmac et SIVcpz (macaque et chimpanzé).

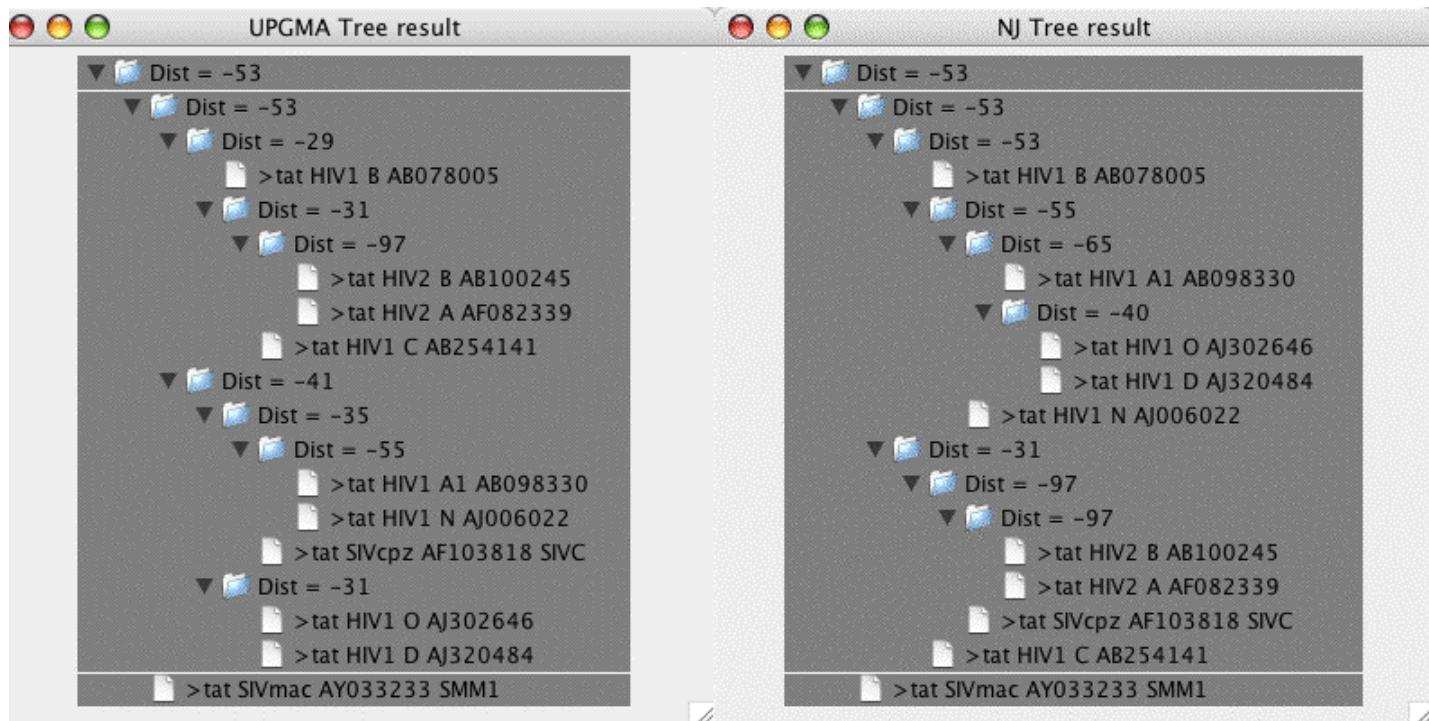
Pour chacun de ces variants du virus, les tests ont été réalisés avec la séquence d'une protéine d'enveloppe de la capsid du virus (Env), la séquence de la protéine tat, les tronçons 1-100, 101-200, 201-300 et 301-400 de la protéine Env.

2. Résultat des tests

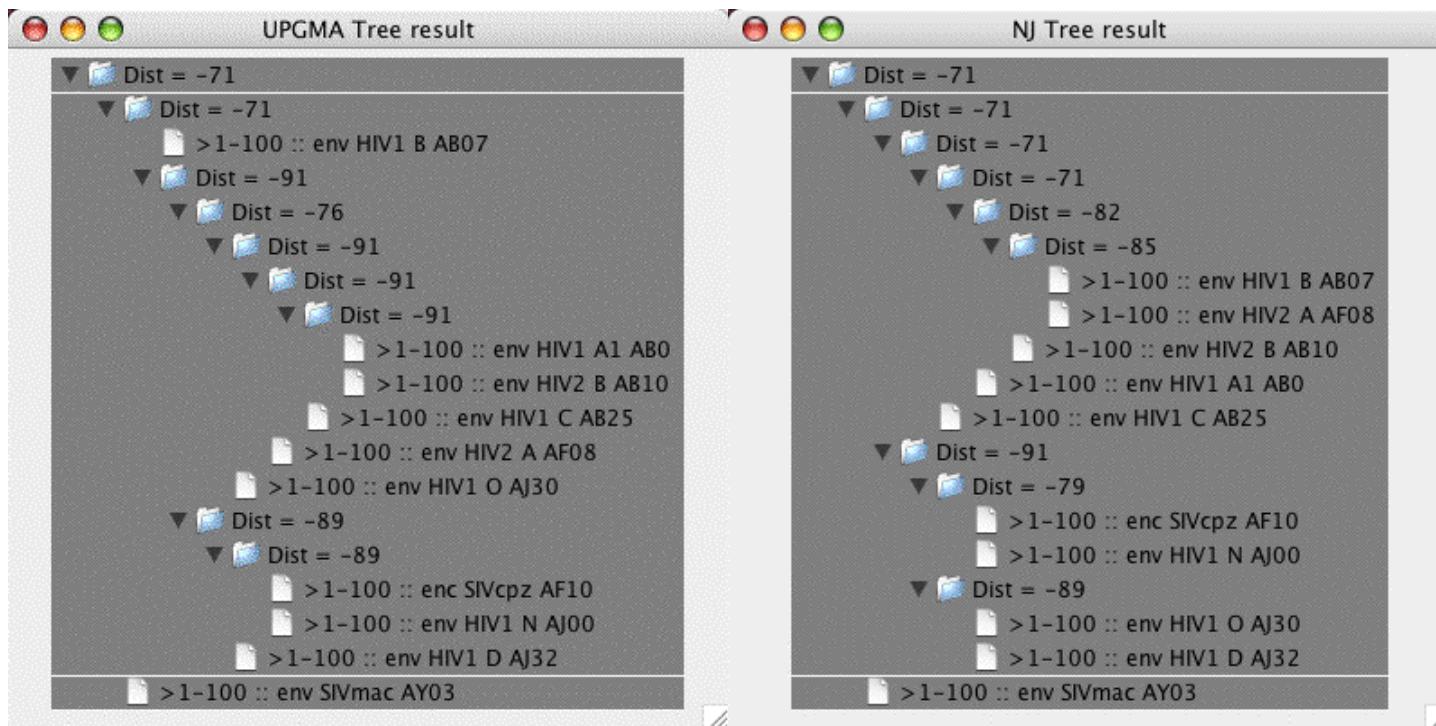
Les arbres obtenus sont les suivants - les valeurs indiquées pour chaque nœud sont les scores obtenus dans la matrice de scores.



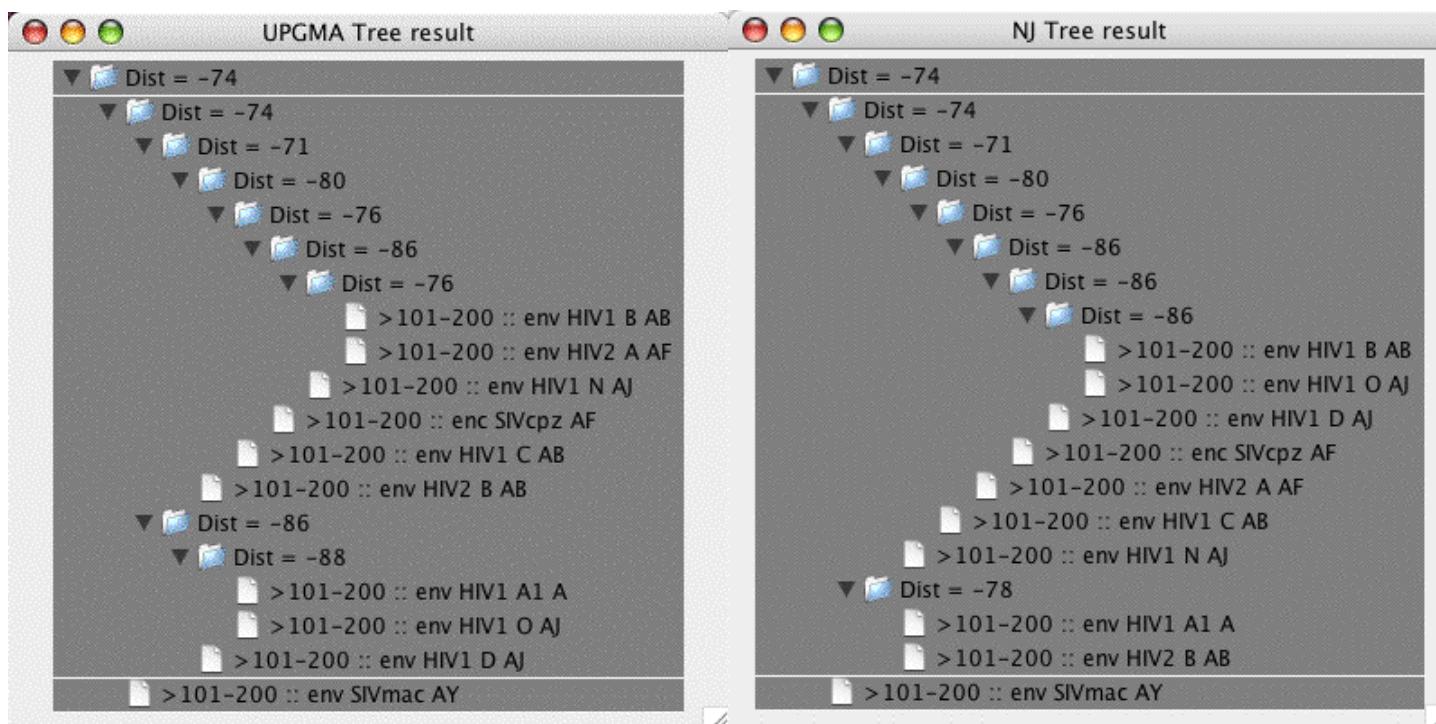
FIGURES 3 ET 4 : ARBRES PHYLOGENETIQUES POUR LA PROTEINE ENV (UPGMA ET NJ).



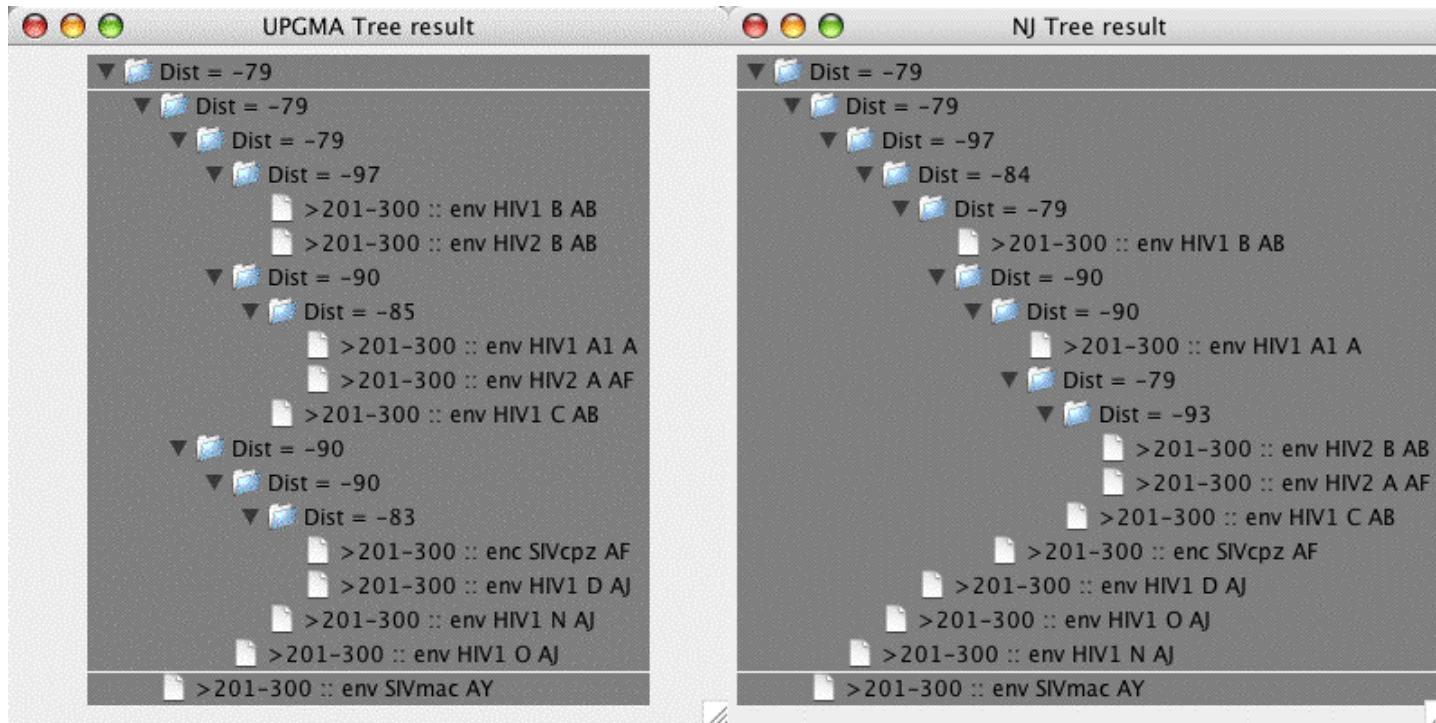
FIGURES 5 ET 6 : ARBRES PHYLOGENETIQUES POUR LA PROTEINE TAT (UPGMA ET NJ).



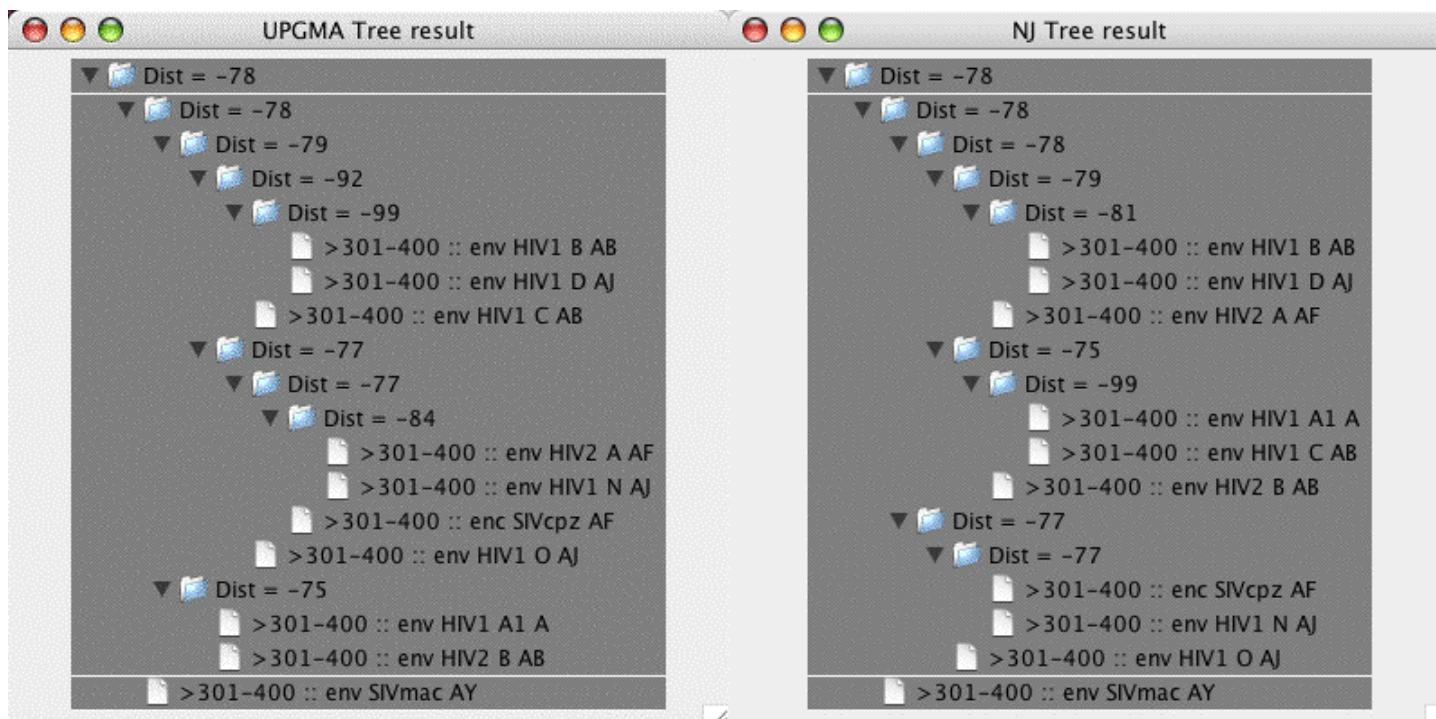
FIGURES 7 ET 8 : ARBRES PHYLOGENETIQUES POUR LA PROTEINE ENV RESIDUS 1 A 100.



FIGURES 9 ET 10 : ARBRES PHYLOGENETIQUES POUR LA PROTEINE ENV RESIDUS 101 A 200.



FIGURES 11 ET 12 : ARBRES PHYLOGENETIQUES POUR LA PROTEINE ENV RESIDUS 201 A 300.



FIGURES 13 ET 14 : ARBRES PHYLOGENETIQUES POUR LA PROTEINE ENV RESIDUS 301 A 400.

• Conclusion et améliorations possibles

L'intérêt du langage Java est sa lisibilité et sa portabilité sur différents systèmes, ce qui confère aux programmes réalisés avec ce langage un caractère réutilisable. Le code du logiciel Phylogenetik est construit pour pouvoir être réutilisé et amélioré.

Quelques améliorations possibles du logiciel sont les suivantes - réalisables lors de projets ultérieurs :

- Réaliser les arbres d'une façon plus classique lors de leur affichage (points et traits pour les nœuds et les branches),
- Calculer les distances à partir de la matrice de scores d'alignements et les intégrer aux branches de l'arbre,
- Implémenter d'autres méthodes de construction d'arbres,
- Enlever de la mémoire courante une ou plusieurs séquences introduites précédemment pour la construction d'arbre,
- Enregistrer la configuration en cours (liste des séquences et arbres construits) dans un format qui puisse être repris par le logiciel.

• Bibliographie et ressources

- [1] Image de couverture : Arbre de classification des espèces de Haeckel, 1866 (Encyclopédie Wikipedia en français). <http://fr.wikipedia.org/wiki/Phylogénie>
- [2] <http://www.info.univ-angers.fr/pub/gh/Idas/Wphylog/infobiogen/phylogenie.htm>
Phylogénie, descriptif et fonctionnement.
- [3] <http://www.biani.unige.ch/msg/teaching/evolution.htm>
Phylogénie, éléments de base.
- [4] <http://philippe.gambette.free.fr/SCOL/NotesBioinfo/node1.html>
Table des matières de notes de cours de bio-informatique.
- [5] http://www.hiv.lanl.gov/components/hiv-db/combined_search_s_tree/search.html
HIV Sequence database.
- [6] <http://bioweb.pasteur.fr/>
Logiciels pour la biologie (Institut Pasteur).
- [7] <http://mesquiteproject.org/>
Mesquite, a modular system for evolutionary analysis.
- [8] <http://www.dina.dk/~sestoft/bsa/bsapplet.html>
Java biosequence alignment applet.
- [9] <http://www.dina.dk/~sestoft/bsa/Match7Applet.html>
Construction of phylogenetic trees.

. Annexes

Un cédérom comportant l'ensemble des données sur ce projet est disponible aux adresses internet suivantes :

<http://gabriel.chandesris.free.fr/ftp/PhylogenetikCD.zip>

<http://gabriel.chandesris.free.fr/projets/Phylogenetik/>

Ces annexes comportent notamment :

- Javadoc de l'ensemble du programme,
- Le code source (format projet Eclipse 3.0),
- Le logiciel Phylogenetik fonctionnel : archive Java (.jar) et une application Mac OS X,
- Jeu d'essais fonctionnel (séquences utilisées pour les exemples),

Ces annexes sont rassemblées avec ce mémoire sur un cédérom directement utilisable sous Windows, Linux et Mac OS X.