Des séquences moléculaires à l'arbre de la vie : résultats théoriques, algorithmes et outils pour la phylogénomique

Vincent Ranwez

▶ **To cite this version:**

Vincent Ranwez. Des séquences moléculaires à l'arbre de la vie : résultats théoriques, algorithmes et outils pour la phylogénomique : Résultats théoriques, algorithmes et outils pour la phylogénomique. Sciences du Vivant [q-bio]. Université Montpellier 2 (Sciences et Techniques), 2010. tel-02824963

HAL Id: tel-02824963
https://hal.inrae.fr/tel-02824963

Submitted on 6 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Montpellier II

Ecole doctorale I2S

Information Structures et Systèmes

Habilitation à Diriger des Recherches (H.D.R.)

Spécialité : informatique

Des séquences moléculaires à l'arbre de la vie :

résultats théoriques, algorithmes et outils pour la phylogénomique

par

Vincent Ranwez

soutenue le 6 décembre 2010 à 14h devant le jury composé de :

M. Mathieu Blanchette	Professeur associé, Université McGill – Canada –	*Rapporteur*
Mme Céline Brochier	Maître de conférences, Université de Provence (H.D.R)	*Rapporteur*
M. Manolo Gouy	Directeur de recherche, CNRS	*Rapporteur*
Mme Hélène Touzet	Directeur de recherche, CNRS/INRIA	*Rapporteur*
M. Emmanuel Douzery	Professeur, Université Montpellier II	*Examinateur*
M. Nicolas Galtier	Directeur de recherche, CNRS	*Examinateur*
M. Olivier Gascuel	Directeur de recherche, CNRS	*Examinateur*

Remerciements

L'art c'est "moi", la science c'est "nous".

Claude Bernard

Les travaux décrits dans ce mémoire sont le fruit de nombreuses collaborations et interactions, à ce titre, ce sont plus « les nôtres » que « les miens ».

Je tiens à remercier Manu pour son accueil au sein de l'équipe phylogénie moléculaire. Malgré son emploi du temps plus que chargé, il a toujours su prendre du temps pour discuter avec moi et a ainsi fortement contribué à mon intégration au sein de l'ISEM. J'ai la chance de travailler dans une équipe dynamique avec Manu, Fred, Nicolas, Sylvain, Julien, François et Marie-ka que j'apprécie tant humainement que scientifiquement. Merci à Vincent Berry, Michel Crampes, Jacky Montmain de m'avoir fait confiance pour co-encadrer des thésards ainsi qu'à Vincent Daubin, Eric Tannier, Laurent Duret, collègues Lyonnais avec qui je collabore depuis plusieurs années.

De nombreux étudiants ont également contribué aux travaux présentés dans ce manuscrit que ce soit lors de projets, de stages ou de thèses. J'ai une pensée particulière pour Céline, la première thésarde que j'ai co-encadrée et qui est maintenant une collègue ; pour Jonathan, Mohameth et Hali dont je co-encadre actuellement les thèses ainsi que pour Nicolas et Sébastien que j'ai eu plaisir à encadrer lors de projets et de stages et qui poursuivent leur chemin au sein de l'EMA.

La charge d'enseignement d'un jeune maître de conférences peut peser lourdement les premières années. Je suis particulièrement reconnaissant à Isabelle Mougenot de m'avoir accompagné lors de ces premières années ainsi que lorsque j'ai assuré la responsabilité de la spécialité Bioinformatique en Master. J'ai la chance de travailler au quotidien avec Isabelle, Marianne, Séverine, Philippe, Jean-François, Mathieu, Anne-Muriel, Annie, Thérèse et Vincent, des collègues qui forment des équipes pédagogiques remarquables.

Je remercie également Nicole Pasteur et Jean-Christophe Auffray qui malgré la lourde tâche de diriger l'ISEM ont pris le temps de me recevoir, de discuter avec moi et m'ont confié des responsabilités au sein de l'ISEM. Le fait d'être un bio-informaticien au sein d'une unité de Biologie est pour moi un atout et une source de satisfaction et d'enrichissement intellectuel, mais c'est parfois une situation relativement complexe et floue au niveau administratif. Merci à J.C. Auffray, O. Gascuel, J.C. Konig, B. Godelle, M. Joab, N. Pasteur, C. Dony d'avoir pris le temps de me rencontrer et de m'avoir éclairé dans mes choix.

Merci à Hélène Touzet, Manolo Gouy, Mathieu Blanchette et Céline Brochier qui ont accepté d'être rapporteurs de ce manuscrit et à l'ensemble des membres du jury avec qui les échanges ont été riches pendant et après ma soutenance.

Merci à Camille et Mathilde, mes deux filles, qui arrivent à me faire oublier mes soucis avec un simple sourire et merci à Sylvie d'être telle qu'elle est et de m'aimer tel que je suis ...

Table des matières

Introduction

Ce mémoire décrit l'essentiel des recherches en bio-informatique que j'ai menées et encadrées depuis mon recrutement (en février 2005) à l'Institut des Sciences de l'Evolution de Montpellier (l'ISEM). En effet, après une formation initiale en informatique (diplôme d'ingénieur, DEA, doctorat) et deux années en tant qu'ingénieur de recherche dans une start-up bio-pharmaceutique, j'ai eu la chance d'être recruté comme maître de conférences au sein de l'équipe Phylogénie Moléculaire (PhylMol) de l'ISEM. Cette affectation a fortement influencé à la fois mes travaux de recherche et ma manière de les publier. Le fait de côtoyer quotidiennement des biologistes moléculaires me permet d'identifier certains des verrous scientifiques auxquels ils sont confrontés est de concentrer mes efforts de recherche en informatique sur ces points clefs. Cette position de bio-informaticien dans une équipe de biologistes moléculaires me permet également d'affiner la pertinence biologique des solutions algorithmiques que je propose. Entouré d'utilisateurs potentiels, je bénéficie également de retours forts utiles lorsque mes travaux débouchent sur des outils que je souhaite mettre à la disposition de la communauté scientifique.

L'équipe PhylMol, dirigée par Emmanuel Douzery, entretient depuis longtemps des collaborations fructueuses avec le Laboratoire d'Informatique de Robotique et de Micro-électronique de Montpellier (le LIRMM). Des liens étroits existent notamment avec l'équipe MAB (Méthodes et Algorithmes pour la Bioinformatique) qui est dirigée par mon ancien directeur de thèse : Olivier Gascuel. Ainsi, lors de mon arrivée a l'ISEM, les membres de l'équipe PhylMol avaient déjà l'habitude de collaborer avec des informaticiens. Tandis que de mon côté, je venais de travailler pendant deux ans avec des biologistes d'un laboratoire privé. Mon intégration au sein de cette équipe, s'est donc faite de manière naturelle. Ma présence dans l'équipe PhylMol a renforcé nos liens avec l'équipe MAB, notamment au travers de nombreuses collaborations avec Vincent Berry, que ce soit pour la rédaction d'articles, les réponses à des appels d'offres, ou le co-encadrement de thèses. Je suis pleinement satisfait de mon intégration au sein de l'ISEM et mes collaborations avec le LIRMM m'ont permis de trouver un équilibre, que j'espère arriver à conserver, entre des travaux très algorithmiques et une recherche plus directement appliquée.

Mes travaux de recherche portent essentiellement sur la phylogénie moléculaire dont l'objectif est de retracer l'histoire évolutive d'espèces vivantes à partir de séquences moléculaires issues de leurs génomes. Ces phylogénies moléculaires fournissent un cadre évolutif nécessaire à de nombreuses études de biologie comparative. En effet, la parenté évolutive des espèces induit de fortes ressemblances entre elles. La phylogénie des espèces étudiées doit donc être prise en compte afin d'évaluer correctement les corrélations existantes entre certains de leurs caractères (e.g. poids du corps/poids du cerveau chez les mammifères, nombre de chromosomes et com-

portements sociaux des insectes, etc.) ou l'impact que peuvent avoir sur elles des facteurs environnementaux (Felsenstein, 1985; Carvalho et al., 2005).

L'analyse de phylogénies moléculaires permet également d'effectuer des études épidémiologiques. Par exemple, dans les années 90, le *center for disease control* d'Atlanta, a reçu un rapport surprenant concernant une jeune femme séropositive. D'après ce rapport, le seul lien entre cette patiente et le virus était d'avoir consulté un dentiste porteur du virus. Après enquête, il s'est avéré que d'autres patients de ce dentiste avaient, eux aussi, contracté le virus du SIDA. Le *center for disease control* a donc réalisé une analyse moléculaire des souches du virus présentes chez le dentiste, chez ses patients, et chez d'autres malades n'ayant jamais consulté ce dentiste. La phylogénie moléculaire de ces souches virales a permis de confirmer que le dentiste avait effectivement contaminé ses patients. Des précautions sanitaires supplémentaires ont donc pu être mises en place pour éviter ce type de contamination (Hillis and Huelsenbeck, 1994; Hillis et al., 1994). Le positionnement d'une nouvelle souche virale (e.g. une nouvelle souche grippale) dans une phylogénie permet également d'identifier les souches apparentées. Les traitements utilisés sur ces souches proches constituent potentiellement de bonnes bases pour le traitement thérapeutique de la nouvelle souche.

La phylogénie moléculaire est aujourd'hui en train de changer d'échelle. Les progrès technologiques nous permettent désormais d'avoir facilement accès aux génomes complets d'un nombre toujours plus grand d'espèces. On parle maintenant de "phylogénomique" pour désigner les études phylogénétiques qui se basent sur l'analyse de génomes complets. Parallèlement, le nombre d'espèces étudiées augmente lui aussi très rapidement, le but en ligne de mire (même si on le sait inatteignable) étant de proposer la phylogénie de l'ensemble des êtres vivants : "l'Arbre de la Vie". Cet objectif est avant tout un défi intellectuel qui reflète l'envie intarissable de l'homme de connaître et de comprendre le monde qui l'entoure. Mais cette connaissance, toujours meilleure, de l'Arbre de la Vie est également un outil clef pour mieux protéger ce monde fragile. En effet, la préservation des espèces est un enjeu majeur de notre siècle. La phylogénie moléculaire fournit un cadre théorique permettant une définition formelle de la notion, souvent floue, de biodiversité (Faith, 1992; Moulton et al., 2007). L'obtention d'un Arbre de la Vie (même partiel) est donc un outil précieux qui permet de définir de manière objective les espèces (ou les zones) à protéger prioritairement (Bininda-Emonds et al., 2000; Forest et al., 2007).

Les travaux présentés dans ce manuscrit s'inscrivent dans ce grand projet qu'est l'inférence de l'Arbre de la Vie. Le premier chapitre de ce mémoire constitue une introduction à la bio-informatique et plus spécialement à la "phylogénie moléculaire". Le deuxième chapitre présente mes travaux sur l'alignement de séquences orthologues. Ces alignements sont à la base des analyses phylogénétiques, il est donc crucial qu'ils soient de bonnes qualités. Nous avons développé une base de données centrée sur ces alignements qui nous a notamment servi à mieux comprendre la structure "en isochore" des génomes mammaliens. Nous avons également développé

un algorithme capable de produire des alignements de bonne qualité même pour des séquences atypiques (e.g. "pseudogénisées"). Le chapitre 3 décrit mes travaux sur les méthodes de "super-arbre" qui permettent de combiner plusieurs phylogénies partielles en une phylogénie plus complète. Il introduit notamment deux propriétés théoriques importantes qui sont souhaitables pour assembler un Arbre de la Vie fiable. Suite à ces travaux, nous avons développé la méthode *PhySIC_IST* qui respecte ces propriétés et qui nous a permis de mieux comprendre la phylogénie d'un groupe de céréales sur lesquels il existe de forts enjeux alimentaires et économiques. Enfin, le chapitre 4 présente des méthodes permettant d'intégrer dans les analyses phylogénétiques des arbres ayant plusieurs représentants de la même espèce (obtenus à partir de "familes multigéniques"). Le fait de construire l'Arbre de la Vie sans prendre en compte ces données a été fortement critiqué, car celles-ci représentent une part importante de l'information disponible. Ce travail est l'un des premier à permettre d'exploiter cette source d'information, qui est pourtant essentielle.

Ces différents travaux sont autant d'éléments qui contribuent à rendre possible une inférence fiable de l'Arbre de la Vie.

Chapitre 1

Phylogénomique : contexte et objectifs

Les ordinateurs sont inutiles. Ils ne savent que donner des réponses.

Pablo Picasso

Sommaire

Je n'ai évidemment pas la prétention de retracer l'histoire complète de la bio-informatique, ni même de décrire l'ensemble des sous domaines qui la composent. J'essaierai simplement dans ce chapitre de fournir quelques éléments clés permettant de comprendre les enjeux de cette discipline, de positionner mes travaux dans ce contexte, et de présenter brièvement la manière dont j'envisage mon travail de bio-informaticien.

1.1 Bio-informatique et phylogénie : définitions

1.1.1 Bio-informatique, kesako ?

Définir ce qu'est la bio-informatique n'est pas une chose simple, plusieurs définitions alternatives existent, qui reflètent à la fois la diversité de cette thématique et sa rapide évolution. Voici quelques-unes de ces définitions.

- La bio-informatique est « l'étude des procédés informatiques dans les systèmes biotiques. » Traduction de la définition proposée par Hogeweg and Hesper (1978) qui sont apparemment les premiers à avoir utilisé ce terme.
- « La bioinformatique est une discipline récente qui propose et développe des modèles, des méthodes et des outils afin d'analyser l'information biologique disponible et produire de nouvelles connaissances. » Introduction à la bioinformatique (Gibas and Jambeck, 2002).
- « La bioinformatique est le traitement automatique de l'information biologique, sous forme de données. » Site Infobiogene (2005), ce site est désormais fermé.
- « La bio-informatique est un champ de recherche multi-disciplinaire où travaillent de concert biologistes, informaticiens, mathématiciens et physiciens, dans le but de résoudre un problème scientifique posé par la biologie. » Wikipédia (2010)
- « bio-informatique (nom féminin). Application de la recherche en informatique au progrès des connaissances dans les sciences de la vie. La bio-informatique constitue une branche nouvelle des sciences biologiques qui s'est développée à partir du début des années 1990, avec les opérations de séquençage du génome humain. À l'interface de la biologie, des mathématiques et de l'informatique, elle consiste principalement en l'acquisition, la mise en forme et l'interprétation de l'information génétique portée par l'A.D.N. des gènes et par les protéines. » Encyclopédie Larousse en ligne (2010).

Le terme bio-informatique est lui-même assez récent, il ne fait son apparition dans le Petit Robert qu'en 2002, et dans le Petit Larousse en 2004. Dans ces deux dictionnaires, l'orthographe retenue est « bio-informatique ». C'est donc celle qui sera utilisée tout au long de ce manuscrit. Bien que ces définitions diffèrent sur de nombreux points, elles soulignent toutes l'aspect pluridisciplinaire de la bio-informatique. Contrairement à ce que son nom suggère, la recherche en bio-informatique est souvent à la croisée de plus de deux disciplines. Les mathématiques y jouent un rôle prépondérant notamment la théorie des graphes, les tests statistiques et les méthodes Bayésiennes. La physique et la chimie y sont également utiles, par exemple pour prédire la structure tridimensionnelle d'une protéine, et leur rôle est capital dans l'une des nouvelles branches de la bio-informatique : la biologie systémique, qui vise à modéliser le fonctionnement d'un système biologique. De même, bien que la bio-informatique soit souvent associée aux données génomiques, ce n'est pas son seul domaine d'application. Outre la biologie systémique, déjà mentionnée ci-dessus, on peut également citer l'analyse d'images (identification automatique de caractéristiques morphologiques), la simulation de l'évolution de la végétation (très utile dans le cas de projets d'urbanisme) ou la recherche d'inhibiteurs d'une protéine dans une base de composés chimiques ("virtual screening"). Enfin, ces définitions se focalisent sur l'analyse de données, qui n'est pas le seul champ de la bio-informatique. En effet, de nombreux travaux portent également sur le stockage et la gestion de ces données (standardisation des formats, thesaurus et ontologies, "web services"), ainsi que sur leurs visualisations (rendu 3D, graphe d'interactions, cartes sémantiques). Je pro-

poserai donc une définition plus large de la bio-informatique : « La *bio-informatique* est un champ de recherche pluri-disciplinaire qui utilise l'informatique pour acquérir, gérer, visualiser, analyser et modéliser des données ou des systèmes biologiques. » L'informatique et la biologie sont les deux seules disciplines explicitement mentionnées dans cette définition, car ce sont celles dont la présence est requise pour pouvoir parler de bio-informatique. Le fait de ne pas lister de disciplines connexes, permet de n'en exclure aucune. Le choix du verbe « utilise » met l'emphase sur le fait que l'informatique est, dans le cas de la bio-informatique, un outil au service du biologiste. Bien entendu, ce "service" peut être indirect, par exemple en facilitant le travail d'autres bio-informaticiens. Mais il me semble, que si les travaux sont conduits sans souci de leurs utilisations finales pour traiter des problèmes biologiques, il ne s'agit pas à proprement parler de bio-informatique. Bien que le théorème de Bayes soit aujourd'hui très utilisé en bio-informatique, il semblerait incongru de dire que le révérend Bayes était un bio-informaticien.

1.1.2 Des classifications fonctionnelles aux phylogénies moléculaires

Parmi les nombreux champs d'application de la bio-informatique, celui qui constitue mon principal champ de recherche est la *phylogénie moléculaire* dont l'objectif est de retracer l'histoire évolutive des espèces à partir de leurs données moléculaires (séquences nucléotidiques ou protéiques). On retrouve dans cette discipline toutes les utilisations de l'informatique mentionnées dans la définition proposée précédemment : acquérir, gérer, visualiser, analyser, modéliser. En effet, une analyse phylogénétique commence par l'acquisition de séquences moléculaires, qui sont ensuite stockées dans des bases de données dédiées que l'on peut interroger à l'aide d'interface Web ou via des langages de programmation. Une fois les séquences d'intérêt collectées, elles sont analysées pour produire un "alignement de séquences" qui peut être visualisé et édité manuellement. Une modélisation du processus d'évolution des séquences moléculaires peut ensuite être utilisée pour rechercher la phylogénie la plus probable étant donné cet alignement de séquences. La phylogénie ainsi obtenue peut à son tour être visualisée, stockée dans une base de données, ou utilisée comme une nouvelle donnée pour d'autres analyses.

Comme en attestent les différents écrits qui nous sont parvenus, les hommes ont toujours essayé d'établir une classification des choses qui les entourent. C'est avant tout une question de survie : il faut être capable de distinguer ce qui est comestible de ce qui est toxique, les plantes qui peuvent soigner de celles qui grattent ou qui piquent. Mais, l'homme ne s'est pas satisfait de ces seules classifications fonctionnelles. Dès l'Antiquité grecque, Aristote propose une classification du vivant qui part du principe que tous les êtres ont une âme, mais une âme de nature et de complexité différente, qu'il utilise pour définir une échelle de la nature : « Par conséquent, pour chaque classe d'êtres, il faut rechercher quelle espèce d'âme lui appartient, quelle est, par exemple, l'âme de la plante, et celle de l'homme ou celle de l'animal » (Aristote, le traité de l'âme, chapitre 3). Au bas de cette échelle du vivant

se trouvent les plantes, dont l'existence n'est, à ses yeux, justifiée que par l'utilisation qu'en font les animaux. Au sommet de cette échelle, évidemment, se trouve l'homme. Pendant longtemps, les plantes ne continueront à être étudiées que d'un point de vue fonctionnel, essentiellement par des médecins, pour des raisons pharmacologiques. Le naturaliste John Ray propose une des premières classifications naturelles des plantes qui distingue notamment les monocotylédones des dicotylédones. Il introduit également une définition du concept d'espèce basée sur la capacité de reproduction (Ray, 1686). Ces travaux sont ensuite complétés par Linné dont la classification des plantes est basée sur leur sexualité et leurs organes reproducteurs. Linné propose également d'attribuer un nom double à chaque espèce vivante, permettant ainsi de nommer tout nouveau spécimen (Linné, 1758). Cette *nomenclature binomiale* (nom de genre/nom d'espèce) est encore utilisée aujourd'hui.

Différentes hypothèses sur le transformisme des espèces ont été proposées dès l'antiquité gréco-romaine. La première réelle tentative d'élaborer une théorie de l'évolution est généralement attribuée à Lamarck. Sa théorie s'opposait de manière directe à la théorie dominante de la préformation et du fixisme des espèces. Cette dernière, fortement soutenue par l'ordre religieux mais aussi par la plupart des scientifiques de l'époque et notamment par Cuvier, est restée la théorie dominante jusqu'à la parution de "L'Origine des Espèces" de Darwin (1859). Sa théorie de l'évolution fournit alors une nouvelle manière d'aborder la classification des espèces. Darwin souligne le lien étroit entre classification et phylogénie : « le lien que nous révèlent partiellement nos classifications, lien déguisé comme il l'est par divers degrés de modifications, n'est autre que la communauté de descendance, la seule cause connue de la similitude des êtres organisés. » Cette vision ajoute une dimension temporelle à la classification et, depuis Darwin, les arbres sont utilisés comme support graphique pour représenter simultanément l'aspect temporel de l'évolution et les groupements d'espèces qui en découlent. La seule illustration de l'origine des espèces est d'ailleurs un arbre, qui souligne le nombre important d'espèces qui ont participé au processus d'évolution et dont la lignée s'est complètement éteinte (Figure 1.1). Cette vision évolutionniste change radicalement la manière d'appréhender la classification, il ne s'agit plus d'établir une classification pratique du vivant, mais de retrouver un ordre naturel intrinsèque. Pourtant les méthodes de classification sont longtemps restées basées sur la comparaison de caractères morphologiques, et ce n'est qu'assez récemment que des classifications prenant explicitement en compte un modèle d'évolution sont apparues. Dans les années soixante, la biologie moléculaire a donné accès aux génomes des espèces et l'apparition des premiers ordinateurs a fourni des outils capables de traiter ces nouvelles données. Plusieurs publications ont alors montré que les données moléculaires permettent de reconstruire des phylogénies cohérentes avec les classifications antérieures (fondées sur l'étude des fossiles et des caractères morphologiques).

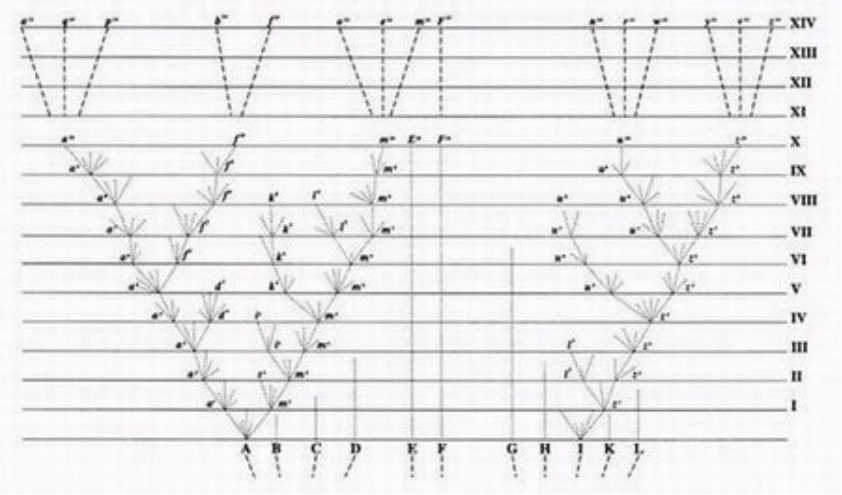

FIGURE 1.1 – **Représentation sous forme d'arbre du processus d'évolution.** *Cette figure est une reproduction de celle que Darwin avait inclue dans son livre : "l'origine des espèces". L'axe vertical représente la dimension temporelle, chaque nœud représente une espèce et les embranchements des événements de spéciation.*

1.2 Séquences moléculaires et phylogénie

On sait aujourd'hui que l'ADN (l'Acide DésoxyriboNucléique), est le support de l'information génétique et de la transmission héréditaire. L'ADN est donc, à l'évidence, une source privilégiée d'information pour la reconstruction phylogénétique. Cette section présente rapidement les avantages liés à l'utilisation de données moléculaires en phylogénie. Elle décrit ensuite de manière plus précise ce que sont les séquences d'ADN, d'ARN et d'acides aminés.

1.2.1 Intérêts des séquences moléculaires en phylogénie

L'inférence de phylogénies s'est longtemps basée sur la similitude de caractères morphologiques et anatomiques (tailles et formes de différentes parties externes ou internes du corps) ou sur des similitudes de traits d'histoire de vie (espérance de vie, mode de nutrition, type de vision). Le problème de cette approche est que, confronté à un même problème, la même solution peut être sélectionnée plusieurs fois. Dans ce cas, la similitude qui en découle ne reflète pas une parenté commune, un trait caractéristique partagé et transmis par un ancêtre commun, mais une adaptation identique en réponse à une pression sélective similaire. En regardant les séquences moléculaires, qui contiennent la manière précise dont la solution retenue est codée sous forme de séquences moléculaires, on peut plus facilement différencier ces deux cas. Le cas de la phylogénie des taupes est particulièrement marquant. En effet la phylogénie moléculaire a permis de mettre en évidence que la taupe dorée avait une relation de parenté plus étroite avec l'éléphant d'Afrique qu'avec la taupe européenne qui lui ressemble pourtant beaucoup plus (Figure 1.2). Finalement, le problème de

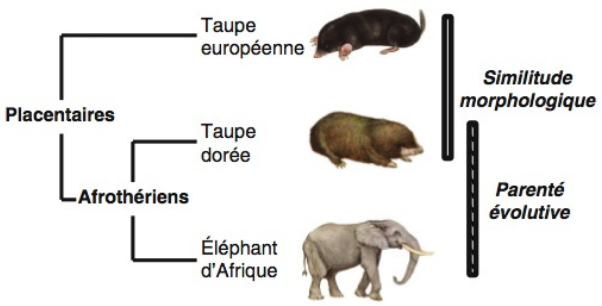

FIGURE 1.2 – **Similitude morphologique et parenté évolutive.** *Cette figure, tirée du chapitre "phylogénie moléculaire" (Blanquart et al., 2010) auquel j'ai contribué, montre à quel point les similitudes morphologiques peuvent être trompeuses.*

retracer l'histoire évolutive des espèces est assez proche de ce que l'on cherche parfois à faire pour évaluer des TP d'informatique. Quand on exécute le programme de deux étudiants différents, et que leurs interfaces sont similaires, que les fonctionnalités implémentées et les bugs sont sensiblement les mêmes, on se demande s'il n'y a pas eu de partage de code. Dans ce cas, à un moment l'un des étudiants donne son code à l'autre, puis chacun continue à le modifier et à l'enrichir. La question est donc de savoir si les deux programmes découlent ou non d'un même programme "ancestral". En cas de doute lors de l'exécution de ces programmes, un réflexe naturel est de consulter leur code source. En regardant ces codes, on dispose de beaucoup plus d'informations qu'en regardant uniquement le résultat de leurs exécutions. Il devient alors plus facile de savoir si la même solution a été trouvée deux fois, ou si ces solutions partagent une histoire commune. D'une certaine manière c'est une simple question de probabilité. Chaque résultat peut être produit par énormément de codes différents, alors que le même code produit le même résultat. Il est donc plus probable de voir apparaître "par hasard" le même résultat que de voir apparaître le même code.

L'utilisation de séquences moléculaires a plusieurs autres avantages. Elles sont notamment simples à manipuler. En effet, les séquences moléculaires d'ADN peuvent être représentées de manière schématique par des mots n'utilisant qu'un alphabet de quatre lettres. De ce fait, il est également relativement simple de modéliser leurs évolutions en prenant en compte les mutations aléatoires qui peuvent apparaître lors de l'auto-réplication de l'ADN. De plus, alors qu'il est déjà compliqué de collecter une centaine de caractéristiques morphologiques différentes, on dispose souvent de séquences moléculaires de plusieurs milliers de caractères ce qui augmente la capacité à différencier l'héritage des convergences aléatoires. Et ce d'autant plus, que les caractères moléculaires sont relativement indépendants les uns des autres contrairement aux caractères morphologiques qui sont souvent fortement corrélés les uns aux autres.

1.2.2 Différents types de séquences moléculaires

L'Acide DésoxyriboNucléique (ou *ADN*) contient quatre types de bases différents. Chaque base est associée à un sucre pour former un *nucléotide* et ces nucléotides sont liés entre eux par des liaisons phosphates constituant ainsi une longue chaîne. Les quatre *bases* présentes dans l'ADN sont regroupées en deux familles distinctes : d'une part les purines, adénine (A) et guanine (G), d'autres part les pyrimidines, cytosine (C) et thymine (T). Ces bases sont complémentaires. Chaque base pyrimidique est liée à une base purique : l'adénine à la thymine et la guanine à la cytosine. L'ADN est constitué de deux chaînes polynucléotidiques complémentaires reliées par des liaisons hydrogènes. La connaissance de la séquence nucléique d'une de ces deux chaînes permet d'établir la séquence de l'autre chaîne. L'Acide RiboNucléique (ou *ARN*) est un polynucléotide proche de l'ADN. On retrouve les deux familles de bases dans la composition de l'ARN, mais la thymine est remplacée par l'uracile (U) qui est une autre pyrimidine. L'ARN est obtenu à partir de l'ADN en utilisant la complémentarité des bases. La *transcription* de l'ADN crée une séquence d'ARN complémentaire à la séquence d'ADN initiale. Il existe plusieurs sortes d'ADN (nucléaire, mitochondrial, chloroplastique) et d'ARN (messager, de transfert, ribosomal, ...). Un *ARN messager*, est un ARN transcrit à partir d'un gène protéique qui est destiné à être traduit en protéine. Les *acides aminés* sont les constituants de base des protéines ; chaque protéine est définie par une séquence particulière d'acides aminés. Il existe 20 acides aminés, chacun étant codé par un mot de trois nucléotides (un *codon*) de l'ARN messager. Il existe au moins un codon pour chaque acide aminé, mais plusieurs codons peuvent représenter le même acide aminé, et certains codons ne représentent aucun acide aminé (codons STOP). Ce *code génétique*, qui permet de traduire les codons en acides aminés, est dit "universel" (même si quelques espèces utilisent en fait des codes légèrement différents).

1.2.3 Une brève histoire du séquençage

Bien que l'accès aux séquences moléculaires soit assez récent, les rapides progrès technologiques en matière de séquençage ont permis d'accumuler très rapidement une quantité phénoménale de données moléculaires. Une des dates clés de cette aventure est l'année 1953 au cours de laquelle Crick et Watson découvrent la structure chimique en double hélice de l'ADN (Figure 1.3). La complémentarité des éléments chimiques qui composent les deux brins de cette hélice permet d'expliquer sa capacité d'auto réplication (Watson and Crick, 1953). La légende veut que cette découverte soit en partie le résultat de discussions passionnées autour de pintes de bières partagées à l'Eagle, un pub de Cambridge (Grande-Bretagne) qui aujourd'hui encore affiche fièrement cette célèbre partie de son histoire (Figure 1.4).

En 1976, le premier génome complet est publié (Fiers et al., 1976), il s'agit du génome d'un virus constitué de trois gènes : le bactériophage MS2. En 1990 le *Department of Energy* et le *National Institute of Health (NIH)* lancent officiellement le projet du séquençage du génome humain. Watson est choisi comme directeur de ce

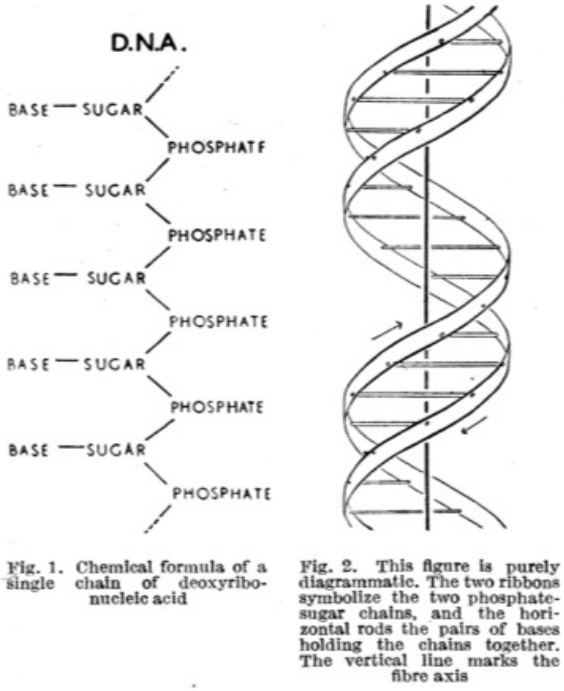

FIGURE 1.3 – **Structure en double hélice de l'ADN.** *Cette figure est tirée de l'article publié en 1953 par Crick et Watson (Watson and Crick, 1953).*

FIGURE 1.4 – **Plaque affiché sur le mur du pub "The Eagle" pour commémorer l'annonce de la découverte de la structure de l'ADN.**

projet. En 1993, apparemment suite à un désaccord sur la politique de brevetabilité des gènes, Watson est congédié et Francis Colins prend la direction de ce projet de séquençage. En 1995 Craig Venter et ses collaborateurs publient le génome de la bactérie Haemophilus influenzae (Fleischmann et al., 1995). C'est le premier génome complet d'un organisme "vivant". En 1998, Craig Venter fonde la société *Celera*, avec pour objectif affiché le séquençage du génome humain. En 2001 les premières analyses du génome humain sont publiées (Lander et al., 2001; Venter et al., 2001). Les revues *Nature* et *Science* y consacrent chacune un numéro spécial. Après avoir découvert que la terre n'était pas le centre de l'univers, que l'homme, comme tout autre animal, était le résultat d'une lente évolution, nous découvrons, en plus, que notre génome contiendrait entre 30 000 et 40 000 gènes, à peine plus que celui d'un ver de terre... Ce léger bémol porte un nouveau coup à notre orgueil, mais n'entame ni l'optimisme des scientifiques, ni celui des journalistes. Dans l'édito du numéro spécial de *Science* consacré au génome humain, on peut ainsi lire : « Humanity has been given a great gift. With the completion of the human genome sequence, we have received a powerful tool for unlocking the secrets of our genetic heritage and for finding our place among the other participants in the adventure of life. » De manière plus pragmatique, un article publié le 13 février 2001 sur le site de RFI (Radio France Internationale) souligne les possibles applications thérapeutiques : « Les scientifiques attendent beaucoup de la connaissance du génome humain pour le traitement et la prévention du diabète, de l'hypertension artérielle, de la maladie d'Alzheimer, du cancer ou encore de la dépendance aux drogues. (...) En isolant des séquences de gènes, il sera plus facile de trouver le gène responsable d'une maladie, de comprendre le mécanisme de l'apparition de cette maladie et d'explorer les solutions thérapeutiques. »

Une difficulté du traitement du séquençage est qu'il ne fournit que de petits fragments du génome (appelés *reads*). Un peu comme si l'on disposait d'un fax capricieux qui n'enverrait à chaque émission qu'une vingtaine de mots consécutifs en partant d'un endroit aléatoire du document et qu'il nous fallait reconstituer le document complet à partir d'une multitude de ces réceptions partielles. Cette reconstruction du document global correspond à l'*étape d'assemblage* du génome. Lorsque la fin d'un fragment du document correspond au début d'un autre on peut supposer qu'ils se suivent et construire un nouveau fragment plus long (un *contig*) en les collant ensemble. Pour la partie commune de ces fragments on dispose alors de deux versions distinctes de la même portion de texte. Le nombre moyen de copies disponibles pour chaque lettre du texte correspond à la *profondeur du séquençage*. Si l'on poursuit l'analogie, on comprend également que la reconstruction du document global (l'assemblage) va être d'autant plus simple que les fragments (les reads) seront plus longs. C'est tout l'enjeux des progrès actuels des technologies de séquençage. A l'inverse, cette reconstruction sera plus compliquée si certaines expressions sont répétées à plusieurs endroits dans le texte (séquences répétées dans le génome) ou si certaines lettres des faxs reçus sont illisibles ou erronées (*erreurs de séquençage*). Il peut également arriver que, malgré de nombreux envois, certaines portions du texte ne soient jamais reçues...

Pour toutes ces raisons, plusieurs années s'écouleront entre la publication en 2001 des premières versions du génome humain et la déclaration officielle en 2004 de la fin de ce projet (International Human Genome Sequencing Consortium, 2004). Le décryptage du génome humain aura pris plus de 10 ans et il aurait coûté près de 3 milliards de dollars. Depuis, les technologies de séquençage ont évolué de manière fulgurante. Il ne faut maintenant que quelques semaines pour obtenir l'assemblage du génome complet d'un organisme. Le coût standard d'un génome est autour de 10 000 $ et de nombreux experts pensent qu'il tombera rapidement sous la barre des 1 000 $.

Le lien entre un gène et une maladie s'est avéré beaucoup plus complexe que prévu. Dans la plupart des cas, c'est tout un réseau de gênes qui est impliqué, les mutations somatiques sur ces gènes ne sont pas des indicateurs binaires mais des facteurs de risques supplémentaires. De nouveaux projets de séquençage voient donc le jour, on peut notamment citer le *1,000 Genome Project* (http ://www.1000genomes.org/) qui vise à séquencer 1 000 génomes humains pour mieux comprendre les liens entre variations génomiques et maladies. Le *Cancer Genome Atlas* est un autre projet important de séquençage dont l'objectif est de comprendre l'évolution (la cinétique) des cancers. En septembre 2009, le gouvernement américain a investi 175 millions de dollars dans ce projet. Parallèlement, un marché se crée pour répondre aux demandes individuelles. Plusieurs sites Internet proposent déjà pour 500 $ de faire une analyse génomique d'un individu. A Stanford, un module intitulé *Genomics and Personalized Medecine* accompagne les étudiants durant l'analyse de leur propre ADN. Dans une interview donnée au journal *USA Today* du 8 juillet 2010, le responsable de ce module précise : « The students will learn troubling things. You will learn that you have a predisposition for lots of different diseases. Some may learn that (...) their father isn't their real father. » Dans le même ordre d'idées, la société *Illumina* propose un service de séquençage aux particuliers, leur site (`http://www.everygenome.com/`) indique que ce service « can provide you with the most comprehensive information on your genome from the world's experts in genetic analysis. Learn about your ancestry, potential response to drugs, and predisposition to health conditions. Uncover your story. » Ainsi tout individu peut obtenir la séquence de son propre génome sur simple demande de son médecin traitant, et la société *Illumina* peut même l'aider à trouver un médecin traitant sensible à ce type d'analyse (Figure 1.5). On peut difficilement reprocher à un individu lambda de placer d'importants espoirs dans l'analyse de son génome quand on sait que Craig Venter, après avoir été licencié de *Celera*, a avoué qu'une grande partie du génome séquencé et publié en 2001 était en fait le sien (voir par exemple l'article, publié le 4 septembre 2007 par le Guardian à ce sujet [1]). Les enjeux médicaux d'une médecine personnalisée tirant parti du génome des patients sont énormes. Les enjeux économiques sont colossaux, tout comme le nombre de questions éthiques qu'une telle médecine soulève.

1. `http://www.guardian.co.uk/science/2007/sep/04/ethicsofscience.internationalnews`

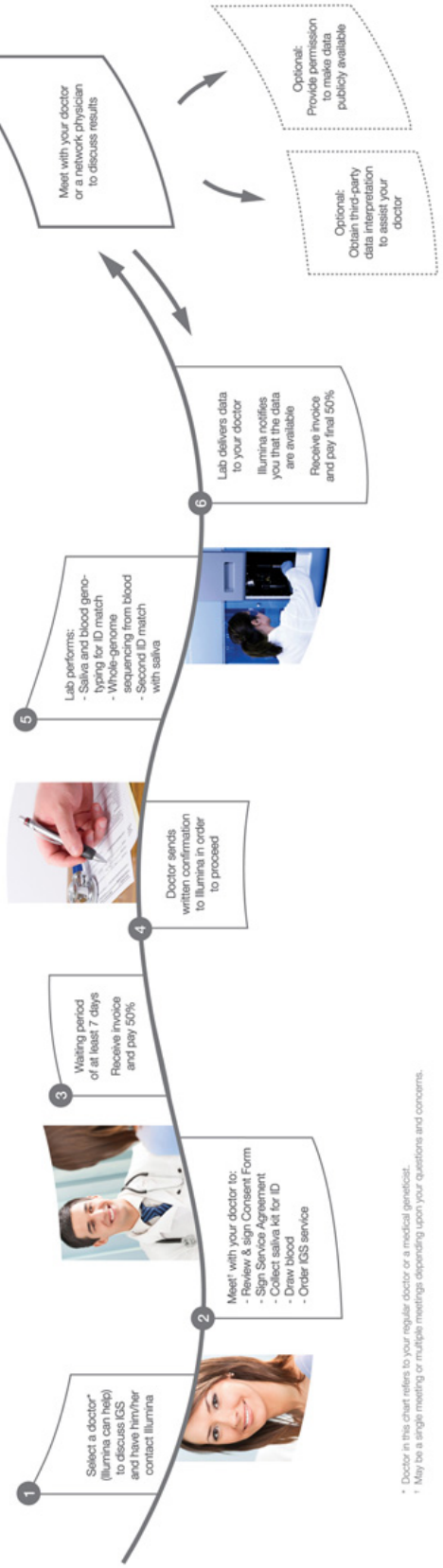

FIGURE 1.5 – **Service de séquençage proposé par la société *Illumina*.** *Ce diagramme, tiré du site Web de la société illumina, illustre la procédure que doit suivre un particulier pour faire séquencer son génome par cette société privée.*

Les progrès en matière de séquençage offrent également de nouvelles perspectives en phylogénie. L'objectif n'est pas simplement de connaître la position exacte d'*Homo sapiens* parmi les primates, mais d'arriver à reconstruire l'histoire évolutive de l'ensemble du monde vivant. Le projet *Tree Of Life* (`http://tolweb.org/`) reflète parfaitement cette volonté de reconstruire la totalité de l'histoire évolutive du vivant. Cet objectif est évidemment utopique. D'une part, parce que l'on ne connaît qu'une faible partie du monde vivant (environ 6 % des espèces (Chapman, 2009)) et d'autre part, parce que certaines parties de cette histoire évolutive se sont passées si vite et sont si complexes qu'il semble irréaliste d'espérer pouvoir les reconstruire sur la base des données moléculaires des seules espèces contemporaines (Rokas and Carroll, 2006). Néanmoins, l'accès aux données moléculaires a considérablement amélioré notre connaissance de cette histoire. Là encore, la réduction des coûts de séquençage a donné lieu à des projets qui vont rapidement nous amener à changer d'échelle. Le site *EnsEMBL* maintient des bases de données autour de génomes complets de vertébré. La version 58 de ce site (mai 2010) s'appuie sur les génomes complets d'une cinquantaine de vertébrés. Le projet *genome10K* (`http://www.genome10k.org/`) a pour ambition de séquencer les génomes de 10 000 vertébrés. Non seulement les coûts de séquençage diminuent rapidement, mais en plus, les techniques s'améliorent. On peut désormais obtenir des séquences de bonne qualité y compris à partir d'un matériel biologique initial de faible quantité et de qualité moyenne. Ces progrès ont permis de séquencer le génome d'espèces éteintes (Hofreiter, 2008) telles que le mammouth (Miller et al., 2008) ou l'homme de Néanderthal (Green et al., 2010). Jusqu'à maintenant, on ne pouvait qu'essayer de prédire ces séquences ; on peut aujourd'hui accéder à ces séquences et, d'une certaine manière, voir le passé.

1.3 Des Séquences moléculaires à "l'Arbre de la Vie"

Cette section décrit brièvement les grandes étapes permettant d'inférer (une partie de) l'Arbre de la Vie à partir de séquences moléculaires.

1.3.1 Inférer l'histoire évolutive d'un gène

Pour pouvoir reconstruire l'histoire évolutive d'un ensemble de séquences moléculaires, il faut que cette histoire existe réellement, et donc que ces séquences soient issues d'une même séquence ancestrale ; on dit qu'elles sont *homologues*. Il arrive que les séquences de certains gènes soient dupliquées chez des organismes. Chaque copie continue alors à évoluer indépendamment de l'autre. Lorsque le but est d'inférer l'histoire des spéciations (la phylogénie), il faut s'assurer que pour chaque espèce on considère bien la même copie du gène. Si c'est bien le cas, on parle alors de séquences *orthologues*.

Les modèles d'évolution des séquences, sur lesquels s'appuient les méthodes probabilistes permettant d'inférer l'histoire de spéciation, se basent sur la probabilité qu'a un nucléotide (ou un acide aminé) d'être muté en un autre en un temps donné.

De même qu’il faut identifier les séquences orthologues, il faut également identifier au sein de ces séquences les nucléotides (resp. acides aminés) orthologues. C’est-à-dire, ceux qui dérivent d’un même nucléotide (resp. acide aminé) ancestral. C’est ce qu’on appelle l’*alignement de séquences*. Le résultat de cet alignement est un nouvel ensemble de séquences obtenues à partir des séquences initiales en leurs insérant un caractère spécial (généralement le "–"). Ces "–" sont insérés de manière à ce que i) les séquences obtenues aient toutes le même nombre de caractères et que ii) les caractères situés à une même position soient orthologues. On représente généralement ces alignements de manière matricielle, chaque ligne contenant la séquence d’une espèce et chaque colonne (appelé *site*) contenant des nucléotides (resp. acide aminé) orthologues. L’orthologie des nucléotides (resp. acides aminés) étant établie, on peut alors inférer l’histoire évolutive du gène.

Le chapitre 2 présente mes travaux en lien avec l’identification et l’alignement de séquences orthologues. Il décrit la base de données OrthoMaM, centrée sur les séquences orthologues chez les mammifères, et résume les résultats biologiques marquants que nous avons obtenus à l’aide de cette base de données. Il explique également comment ce travail a contribué à ce que je m’intéresse récemment à la problématique de l’alignement de séquences codantes et décrit l’approche algorithmique utilisée dans le programme MACSE qui est l’aboutissement de ces travaux.

1.3.2 Inférer l’histoire évolutive des espèces

Chaque gène garde des traces de l’histoire évolutive des espèces qui le portent. Mais ces traces ne permettent généralement de reconstruire qu’une partie de cette histoire. Il faut garder en mémoire que nous partageons plus de 95% de notre patrimoine avec le chimpanzé. Les séquences de certains gènes sont parfois complètement identiques entre plusieurs espèces. Lorsque l’on étudie un ensemble de séquences orthologues, le taux de similitude entre ces séquences est un facteur important qui conditionne la qualité de la reconstruction phylogénétique. En effet, il est impossible de reconstruire la phylogénie de ces séquences si elles sont toutes semblables ou si elles ne partagent aucun point commun. Or cette similitude dépend à la fois du temps depuis lequel ces séquences ont divergé et de leur “vitesse d’évolution”. En effet, la *vitesse d’évolution* d’une séquence (i.e. le nombre de mutations moyen d’un de ces caractères par unité de temps) dépend de “l’importance de ce qu’elle code”, c’est-à-dire des contraintes fonctionnelles qui pèsent sur elle. Les séquences non-codantes sont donc généralement celles qui évoluent le plus rapidement. En fait, elles évoluent souvent de manière si rapide qu’il est difficile d’établir leur orthologie et de les aligner, ce qui explique qu’elles soient assez peu utilisées en phylogénie moléculaire.

On ne peut pas modifier l’époque à laquelle les événements évolutifs ont eu lieu, par contre on peut choisir des séquences qui évoluent plus ou moins rapidement suivant que l’on cherche à reconstruire une phylogénie portant sur des événements récents ou anciens. Il est cependant fréquent que, pour un groupe d’espèces donné, certains événements soient récents et d’autres plus anciens. Il est alors souhaitable

d'utiliser les séquences de plusieurs gènes afin de pouvoir retracer ces deux types d'évènements. Malgré ces précautions, il est possible que certains événements de spéciations n'aient laissé aucune trace sur ces deux gènes, il est également possible que les séquences de ces gènes contiennent, du fait de l'aspect aléatoire des mutations, des similitudes qui ne sont pas dues à une histoire commune (*homoplasie*) et brouillent le signal phylogénétique. Des similitudes trompeuses peuvent également être dues à des événement de duplications et de pertes de gènes ou à l'échange de patrimoine génétique entre deux espèces contemporaines. En effet, certains organismes peuvent transmettre une partie de leur patrimoine à des espèces vivant dans le même environnement. On parle alors de *transfert horizontal* en référence à la représentation arborée de l'évolution où le temps est représenté sur l'axe vertical (Figure 1.1).

Comme nous venons de le voir, il existe plusieurs phénomènes biologiques à cause desquels l'histoire évolutive des séquences ne reflète pas forcément la phylogénie des organismes contenant ce patrimoine génétique. De plus on ne s'intéresse généralement pas à l'histoire de ces organismes particuliers mais plutôt à l'histoire des taxons auxquels ils appartiennent. Un *taxon* est un ensemble d'organismes dont l'unité est admise et qui a des relations de parenté avec d'autres ensembles comparables. On parle de taxon pour désigner indifféremment les souris, les caniches ou une souche particulière d'un virus. Chaque organisme ayant un patrimoine génétique qui lui est propre, les phylogénies reconstruites dépendent en partie des représentants que l'on utilise pour chaque taxon (Douzery and Philippe, 1994).

On peut comparer l'inférence de la phylogénie des espèces à une enquête policière où les gènes des organismes seraient les témoins. Chacun d'entre eux peut témoigner d'une partie différente de l'histoire et certains témoignages, en plus d'être incomplets, peuvent conduire sur de fausses pistes. Pour reconstruire l'histoire complète il est nécessaire de collecter un maximum de témoignages aussi fiables que possibles et de les confronter les uns aux autres. L'inférence de la phylogénie des espèces se fait donc, de plus en plus, en s'appuyant sur de nombreux gènes. Si cette analyse se base non pas sur quelques gènes, mais sur des données couvrant l'ensemble des génomes, on parle alors de *phylogénomique*. Deux grandes familles de méthodes existent pour combiner le signal de plusieurs gènes : les méthodes de *super-matrices* et celles de *super-arbres* (Figure 1.6). Les premières concatènent les séquences des gènes pour produire un "super-gène", et infèrent l'histoire évolutive de ce "super-gène". Les secondes combinent les arbres obtenus à partir de chacun de ces gènes en un "super-arbre". L'approche par super-arbre est notamment privilégiée lorsque les données initiales sont hétérogènes, par exemple dans le cas de données sources d'origine morphologique et moléculaire, nucléotidique et protéique, ou encore représentant la présence-absence d'événements génomiques rares. C'est également une solution permettant d'intégrer dans l'analyse des phylogénies publiées par d'autres auteurs. Enfin, dans certains cas, la taille du "super-gène" est telle que l'approche par super-matrice n'est simplement pas possible.

Une part importante de mes travaux porte sur les méthodes de super-arbres et sur leurs propriétés. Le chapitre 3 résume les résultats que nous avons publiés sur

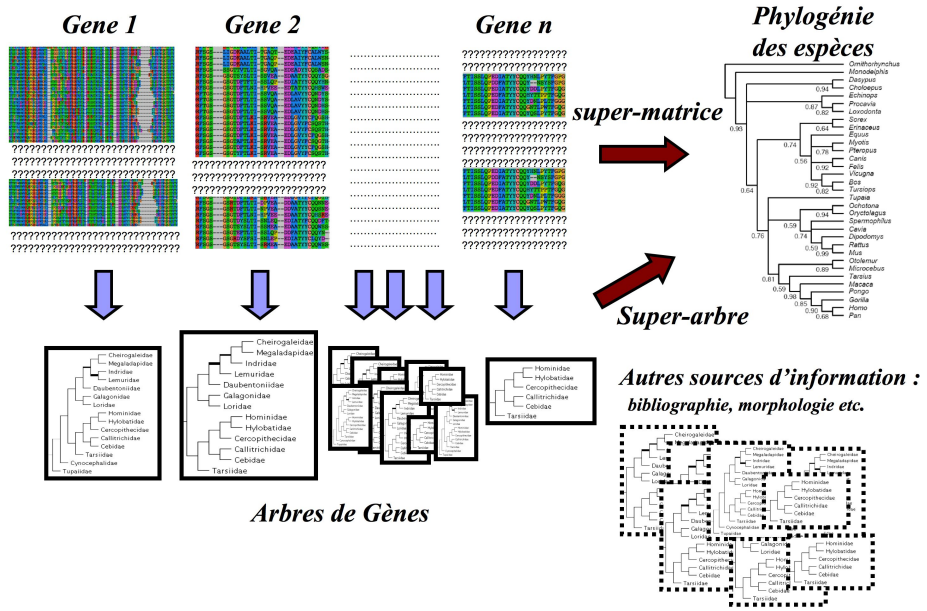

FIGURE 1.6 – **Représentation schématique des approches de super-matrices et de super-arbres**

ce sujet.

1.3.3 Prendre en compte les évènements de macro-évolution

Que l'on utilise une approche de type super-matrice ou de type super-arbre, dans les deux cas on prend essentiellement en compte les évènements de mutation d'un nucléotide (ou d'un acide aminé) en un autre et l'on essaye de résumer la totalité de l'histoire évolutive par une structure arborée. Cependant, les histoires évolutives des gènes et des génomes sont bien plus complexes que le simple modèle arboré et sont engendrées par un processus à plusieurs échelles : des micro-événements (comme les substitutions) agissent au niveau de chaque site et des macro-événements (comme les duplications ou les transferts horizontaux de gènes) agissent au niveau d'un génome ou entre génomes. L'importance de ces macro-événements est telle que le fait même d'essayer de reconstruire un arbre des espèces a fait l'objet d'un intense débat au sein de la communauté scientifique (e.g. Bapteste et al., 2005; Dagan and Martin, 2006; Doolittle and Bapteste, 2007; Galtier and Daubin, 2008). Une approche par simulation (intégrant les transferts horizontaux) a cependant permis de vérifier la pertinence et la faisabilité de la reconstruction d'un arbre des espèces même en cas de conflits réels entre gènes (Galtier, 2007).

Il n'en reste cependant pas moins vrai que ces macro-évènements posent plusieurs problèmes. En effet, les transferts horizontaux, tout comme les évènements de

duplication de gènes, peuvent conduire à observer plusieurs copies d'un même gène chez certaines espèces (on parle alors de *famille multigénique*). Les arbres inférés à partir de ces gènes contiennent alors plusieurs fois la même espèce (ce sont des *arbres multi-labels*). Or les méthodes de super-arbres actuelles sont incapables de gérer ce type de données qui sont pourtant les plus fréquentes. A titre d'exemple, 70% des arbres présents dans la base de donnée HOGENOM (Penel et al., 2009) sont multi-labels. Bapteste et al. (2008) ont donc, à juste titre, reproché au super-arbre obtenu en les excluant d'être l'arbre représentant une minorité du signal disponible et l'ont qualifié de « 1% tree of life ».

Le chapitre 4 décrit les outils méthodologiques (théoriques et logiciels) que nous avons développés afin d'extraire le signal de spéciation d'un arbre multi-labels. Ces outils nous permettent d'exploiter le signal de spéciation présent dans les familles multi-géniques qui était jusque-là ignorées. Bien qu'encourageante, cette stratégie de transformation d'arbres multi-labels en arbres mono-labels est plus une manière de contourner le problème qu'une véritable solution. Notre réel objectif est de concevoir une méthode de super-arbre capable d'intégrer explicitement les macro-évènements et de produire non seulement un arbre des espèces mais également une annotation de cet arbre indiquant les points chauds de transferts et de duplications. Ce chapitre présente l'état de nos travaux sur ce sujet, il détaille notamment un algorithme permettant d'inférer les évènements de macro-évolution permettant d'expliquer les différences observées entre un arbre de gènes et l'arbre d'espèces correspondant.

L'obtention d'alignements de séquences codantes orthologues, la combinaison de phylogénies partielles en une phylogénie plus complète et la prise en compte des évènements de type macro-évolutifs sont trois étapes clefs nécessaires à l'inférence de l'Arbre de la Vie. Les chapitres suivants présentent mes contributions pour chacune de ces étapes, leur objectif commun étant l'obtention d'un Arbre de la Vie aussi complet et fiable que possible.

Chapitre 2

Alignement de séquences codantes orthologues

Some compilers allow a check during execution that subscripts do not exceed array dimensions. This is a help, but not sufficient. First, many programmers do not use such compilers because "They're not efficient." (Presumably, this means that it is vital to get the wrong answers quickly.)

Kernighan & Plauger - The Elements of Programming Style

Sommaire

La première étape en phylogénie moléculaire consiste souvent à collecter et à aligner un ensemble de séquences codantes orthologues. Dans un premier temps, ce chapitre présente la base de données OrthoMaM (Ranwez et al., 2007b) et les résultats biologiques marquants concernant la conversion génique biaisée que nous avons mis en évidence à l'aide de cette base de données (Galtier et al., 2009; Romiguier et al., 2010). Nous décrivons ensuite une solution algorithmique efficace pour aligner plusieurs séquences codantes d'ADN. Cet algorithme est implémenté dans

le programme MACSE qui, à notre connaissance, est la première solution permettant d'aligner plusieurs séquences codantes en prenant simultanément en compte leurs séquences nucléotidiques et leurs traductions en acides aminés (Ranwez et al., soumis).

2.1 Quelques notions supplémentaires de biologies moléculaires

Afin de pouvoir présenter clairement la base de données OrthoMaM et nos résultats concernant la conversion génique biaisée, il est nécessaire d'introduire quelques notions de biologies moléculaires supplémentaires.

2.1.1 La structure des gènes

Un *gène* correspond à une portion de séquence d'ADN qui code un ARN fonctionnel (tel que l'ARN de transfert qui apporte les acides aminés aux ARN messagers lors de leurs traductions) ou une protéine (par exemple une enzyme). Cette séquence d'ADN peut se situer, selon les gènes, sur l'un ou sur l'autre des *deux brins* de la double hélice d'ADN. De manière schématique, l'ADN est dans un premier temps *transcrit* en pré-ARN messager (ou pré-ARNm). Les parties non codantes (ou *introns*) de ce pré-ARNm sont ensuite éliminées (c'est *l'épissage*) pour produire un ARN messager mature (ou *ARNm*) qui ne contient plus que les parties codantes (ou *exons*). Cet ARNm est ensuite traduit en une séquence d'acides aminés (ou AA) afin de produire la protéine codée par le gène. La *traduction* est effectuée par le ribosome en allant de la *partie 5'* de la séquence vers sa *partie 3'*. Il faut également noter que les exons ne sont pas traduits individuellement "in vivo", il est donc possible qu'un codon soit à cheval sur deux exons. Dans ce cas si l'on traduit imprudemment "in silico" le second exon en considérant que ces trois premières bases codent son premier AA, que les trois suivantes codent son second AA etc., on obtient une traduction complètement erronée de cet exon. Trois traductions différentes sont en fait possibles suivant que l'exon commence par la première, la deuxième ou la troisième base d'un codon. Ces trois traductions correspondent aux trois *cadres de lecture* possibles d'une séquence. Cette vision schématique de la transformation d'une séquence d'ADN en protéine (ADN $\Rightarrow$ ARN $\Rightarrow$ protéine) est souvent appelée le "dogme central de la biologie" en référence à l'expression utilisée par (Crick, 1958, 1970). Même si on sait aujourd'hui qu'il existe des déviations par rapport à ce "dogme" (e.g. Mattick, 2003), il continue de fournir un cadre utile pour comprendre une grande partie de la biologie moléculaire.

Pour la suite de ce chapitre, il est cependant utile de préciser que le début et la fin de l'ARNm ne sont généralement pas traduits. Ces régions non traduites, ou *UTR (pour "UnTranslated Region")*, jouent un rôle important dans le contrôle du nombre de copies de la protéine qui sont produites : le niveau d'expression du gène (Pesole et al., 2000). La traduction de l'ARNm s'arrête lorsque le ribosome

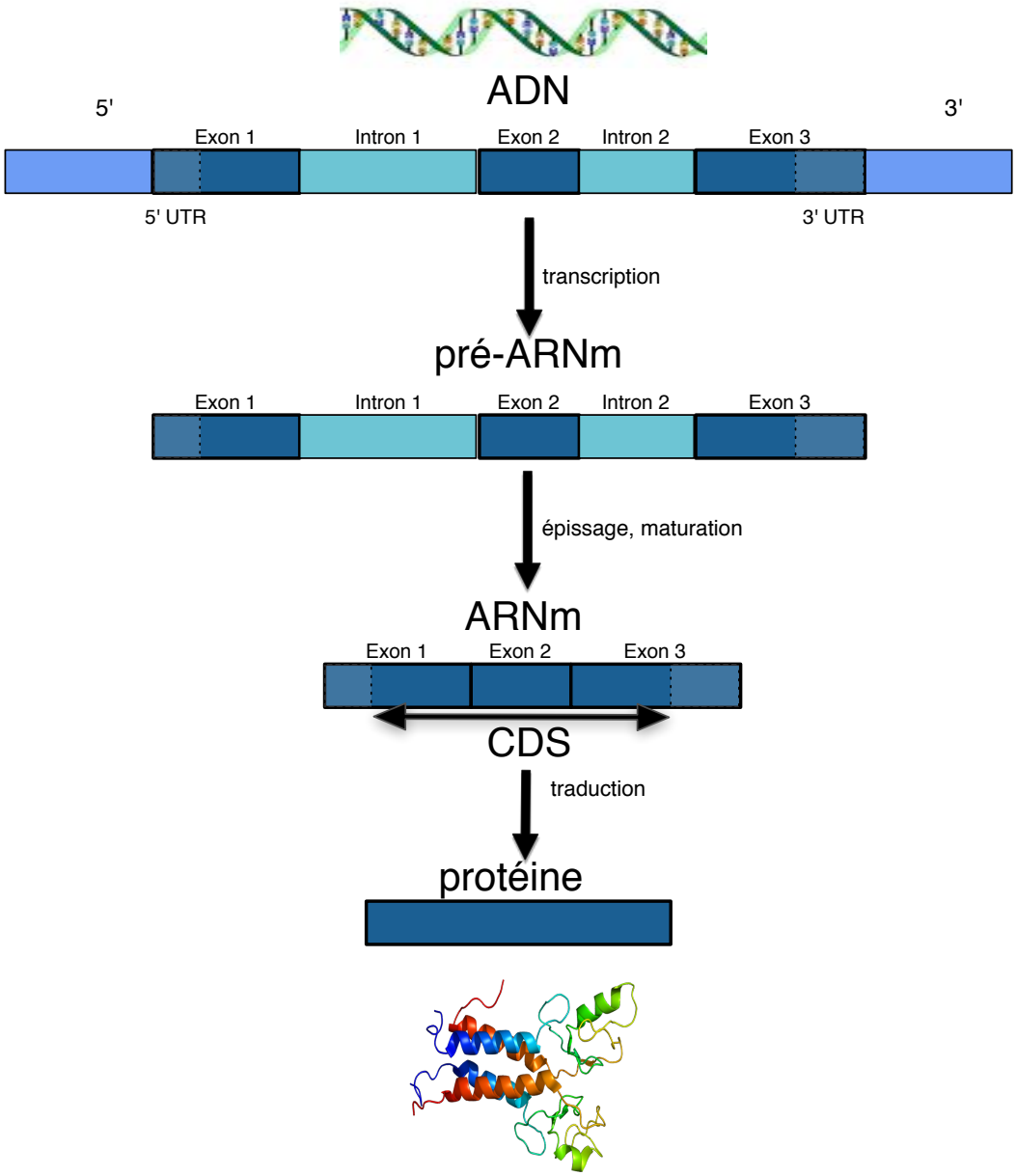

FIGURE 2.1 – **Dogme central de la biologie moléculaire.** *Représentation schématique de la transformation de l'ADN, en protéine.*

rencontre un *codon stop*, la suite de l'ARNm n'est alors pas traduite. La portion d'ARNm qui est effectivement traduite (i.e. privée des UTR) est appelée *CDS* (Co-Ding Sequence). Il est également fréquent d'observer plusieurs épissages possibles de la même séquence d'ADN, on parle alors *d'épissages alternatifs*. Dans ce cas, les exons conservés diffèrent d'un épissage à l'autre, ce qui produit des ARNm différents et donc des protéines alternatives (ou *protéines isoformes*). Ce processus, observé dès 1977, (Chow et al., 1977) était alors considéré comme exceptionnel. En fait l'épissage alternatif semble être la règle plutôt que l'exception. Chez l'homme plus de 90% des gènes ont plusieurs isoformes (Hallegger et al., 2010), la drosophile quant à elle possède un gène ayant plus de 30 000 isoformes ! (Wojtowicz et al., 2004). Ainsi, un gène peut coder plusieurs protéines, ce qui relativise partiellement le fait que le génome humain contienne aussi peu de gènes (cf. section 1.2.3).

2.1.2 Les pseudogènes

Il arrive qu'au cours de l'évolution, une séquence qui codait pour une protéine devienne inutile et ne soit plus traduite. Cela arrive souvent lorsqu'un gène est dupliqué car il existe alors deux gènes qui produisent la même protéine ce qui diminue la pression de sélection. En effet, si une mutation apparaît sur une des copies et empêche la traduction correcte de cette séquence, par exemple du fait de l'introduction d'un codon stop au milieu de la séquence, ce n'est pas très grave puisqu'il existe une deuxième version du gène capable de produire cette même protéine. La séquence altérée n'est alors plus fonctionnelle (elle n'est plus transformée en protéine), elle ne subit plus de pression sélective et peut alors subir de nouvelles mutations induisant des changements de cadre de lecture ou l'introduction de codons stop. Cependant cette séquence conservera encore quelque temps une forte similarité (au niveau AA) avec les séquences fonctionnelles présentent chez les autres espèces, on appelle ce type de séquences des *pseudogènes* (Jacq et al., 1977). La perte de fonctionnalité d'un gène peut aussi survenir lors d'adaptation morphologique. Ce fut par exemple le cas lorsque les dents ont été remplacées par des fanons chez l'ancêtre de certaines baleines. Dans ce cas, les gènes impliqués dans la formation et le développement des dents (comme le gène ENAM) deviennent inutiles et sont *pseudogénisés* (Meredith et al., 2009). L'étude du processus évolutif moléculaire conduisant à ce type d'adaptation morphologique nécessite l'analyse simultanée de séquences fonctionnelles et non fonctionnelles du gène étudié. En particulier, il est important de disposer d'un alignement, qui inclut ces deux types de séquences nucléotidiques et respecte leur structure en codons, pour pouvoir quantifier la variation de pression sélective subit par ce gène (Anisimova and Kosiol, 2009). Cette vision classique où les pseudogènes sont les vestiges inutiles d'anciennes séquences fonctionnelles est actuellement remise partiellement en cause. Une équipe vient en effet de montrer que les pseudogènes, sous forme d'ARN, peuvent interférer avec les mécanismes de régulation de gènes fonctionnels. Un pseudogène pourrait servir de leurre lors de tentative d'inhibition de l'expression d'un gène fonctionnel ayant une séquence très proche. Cette découverte (Swami, 2010), faite dans le cadre de la recherche sur le cancer, renforce l'intérêt

d'analyser ce type de séquence.

2.1.3 Raffinement de la notion d'orthologie

Lorsque l'on considère une famille de gènes homologues (e.g. dérivant d'un même gène ancestral), certains de ces gènes ont été séparés lors d'une spéciation (ce sont des gènes orthologues) tandis que d'autres sont issus d'une séparation liée à une duplication (ce sont des *paralogues*). Si l'on considère le génome de l'homme et celui de la souris (famille des murines), certains gènes de l'homme sont orthologues à plusieurs gènes de la souris tandis que d'autre ne sont orthologues qu'à un seul (Figure 2.2). On dit que deux gènes sont orthologues uniques (ou *orthologues 1 :1*) lorsque chacun d'eux est le seul orthologue de l'autre (pour l'espèce concernée).

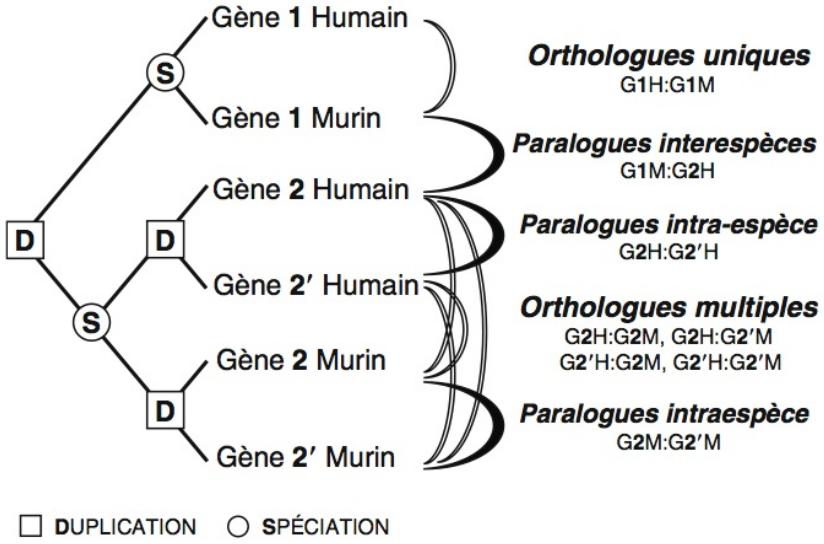

FIGURE 2.2 – **Orthologie versus paralogie.** *Cette figure, tirée du chapitre "phylogénie moléculaire" (Blanquart et al., 2010) auquel j'ai contribué, illustre plusieurs types d'orthologies et de paralogies possibles. L'histoire évolutive de différentes copies humaines (H) et murines (M) des gènes 1 (G1) et 2 (G2) est représentée. Les arcs blancs et noirs connectent respectivement des copies orthologues et paralogues.*

2.1.4 Isochore et conversion génique biaisée

Bernardi et al. (1985) ont montré que les génomes de mammifères et d'oiseaux présentent une forte hétérogénéité en composition de bases nucléotidiques. En effet, ils ont constaté que si l'on découpait un génome en long fragment chromosomiques, le pourcentage de bases G et C observé dans ces fragments (ou *taux de GC*) pouvait varier de 35 à 55%. Ces génomes furent alors présentés comme étant une succession de régions nettement délimitées, ayant chacune sa propre composition en GC. Bernardi donna le nom d'*isochore* à ces régions, et qualifia en conséquence le génome des vertébrés à sang chaud de mosaïque d'isochores. Ce modèle fut confirmé, bien que

nuancé, avec l'avènement des séquençages de génomes complets. En règle générale, il n'y a pas de frontières très nettes entre les régions dont le contenu en GC diffère, mais plutôt une variation continue de la composition en bases. L'analyse des séquences génomiques confirme cependant l'existence de variations significatives du contenu en GC le long des chromosomes. L'analyse du génome humain complet (Lander et al., 2001) a également permis de confirmer que les isochores GC-riches correspondent à des régions riches en gènes et pourvues d'introns courts. Cette prédominance du couple de bases GC est d'autant plus difficile à expliquer que les mutations les plus fréquentes sont des mutations vers les bases A ou T (Webster and Smith, 2004). Il semble que les mutations vers G ou C bien que plus rares que celles vers A ou T, aient une plus grande probabilité de se fixer dans la population (Duret et al., 2002; Spencer et al., 2006). Pourtant celles-ci ne semblent pas apporter d'avantages évolutifs qui pourraient expliquer cela. A la fin des années 70, le livre de Dawkins (1976) et la publication de Orgel et al. (1980) ont bouleversé la vision traditionnelle des génomes au service de l'organisme, avec leur hypothèse de « l'ADN égoïste » : « le parasite ultime ». L'idée centrale exposée dans ces publications est que l'ADN (non codant) peut être sélectionné au cours de l'évolution uniquement pour lui-même et non pour l'organisme qui l'héberge. Un tel mécanisme semble permettre d'expliquer les isochores sans faire appel à la sélection ou à des biais mutationnels variables : la conversion génique biaisée.

La conversion génique se déroule pendant la méiose, lorsque les chromosomes homologues maternels et paternels s'apparient. Il peut alors arriver qu'un segment d'ADN formé d'un brin du chromosome paternel et d'un brin du chromosome maternel soit créé (on appelle cela un *hétéroduplexe*). Cela ne change rien si l'ADN maternel et paternel sont identiques sur cette portion (homozygotie). Mais il est aussi possible qu'ils diffèrent sur cette zone (hétérozigotie). Dans ce cas, l'hétéroduplexe va contenir des mésappariements (un T en face d'un G par exemple). Ceux-ci peuvent êtres reconnus et réparés en utilisant l'un ou l'autre des *allèles* comme modèle, conduisant ainsi à la conversion d'un des deux allèles vers l'autre. C'est ce qu'on appelle la *conversion génique* (Chen et al., 2007b). Si le mésappariement est réparé en favorisant l'un des deux allèles, on parle alors de *conversion génique biaisée* (Figure 2.3). L'allèle favorisé sera plus fréquent et aura plus de chance d'être transmis à la descendance, ce qui augmente sa probabilité de se fixer dans la population. La conversion génique biaisée vers les allèles GC riches est donc une explication plausible à l'existence des isochores (Brown and Jiricny, 1989; Eyre-Walker, 1993). Cette hypothèse est encore plus crédible depuis que l'on sait que les réparations des mésappariements lors des mitoses sont effectivement biaisées vers GC (Birdsell, 2002).

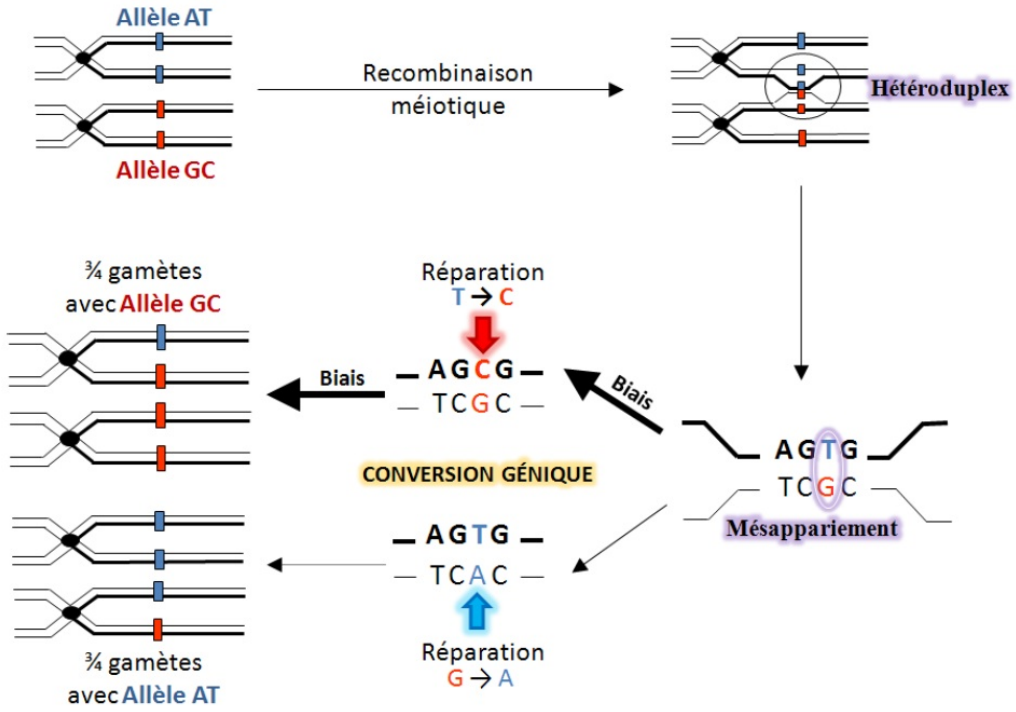

FIGURE 2.3 – **Conversion génique biaisée.** *Cette figure, inspirée de Eyre-Walker (1993), illustre le processus de conversion génique biaisée. Un mésappariement, due à un hétéroduplexe, peut être réparé en utilisant l'un ou l'autre des allèles. Cette conversion génique est dite biaisée si la réparation favorise l'un des deux allèles.*

2.2 OrthoMaM : une base de données de séquences orthologues

2.2.1 Les origines du projet

L'équipe «phylogénie moléculaire» de L'ISEM, travaille depuis de nombreuses années sur la phylogénie des mammifères et plus particulièrement sur celle des rongeurs et des primates. Elle dispose d'échantillons de tissus pour de nombreuses espèces appartenant à ces groupes taxonomiques, et continue régulièrement d'en acquérir de nouveaux. En 2006, au début des travaux sur OrthoMaM, les techniques de séquençage haut débit n'étaient pas encore la norme. Pour résoudre la phylogénie d'un groupe d'espèces, il fallait choisir un ou plusieurs gènes ou fragment de gènes (les « marqueurs phylogéniques ») et acquérir les séquences correspondantes pour l'ensemble des espèces étudiées. Evidemment, si toutes ces séquences étaient présentes dans les banques de données publiques, la collection de tissus n'avait que peu d'intérêt. Elle prenait en revanche toute sa valeur lorsque certaines des séquences n'étaient pas encore disponibles. Il fallait alors les séquencer en interne, à l'aide de « séquenceurs capillaires ». Or cette technique de séquençage nécessite de connaître à peu près correctement le début et la fin de la partie à séquencer (les *amorces*). La séquence n'étant pas encore connue, la solution pragmatique utilisée pour définir ces amorces se basait sur l'alignement des séquences orthologues disponibles. En effet, un tel alignement permet d'identifier les zones suffisamment conservées pour qu'on puisse espérer qu'elles sont également similaires chez l'espèce à séquencer. Ces fragments conservés constituent donc potentiellement des amorces permettant le séquençage chez l'espèce d'intérêt. Ce pari ne fonctionnait pas à tous les coups. Et même lorsque les amorces étaient correctes, le séquençage restait coûteux en temps et en matériel biologique (dans certains cas l'échantillon de tissu est minuscule et précieux). Une fois la séquence obtenue, il arrivait qu'elle ne permette pas de résoudre la phylogénie, il fallait alors tout recommencer avec un autre marqueur.

L'idée initiale du projet OrthoMaM était de faciliter ce processus, en fournissant les alignements de séquences pour de nombreux marqueurs phylogénétiques potentiels (afin d'aider à la création des amorces) et en fournissant pour chacun de ces marqueurs un ensemble de descripteurs (afin d'aider à prédire leurs capacités à résoudre telle ou telle question phylogénétique). L'arrivée des nouvelles techniques de séquençage a partiellement changé la manière dont nous envisageons OrthoMaM. En effet, OrthoMaM se focalise sur les génomes complètement séquencés et assemblés mis à disposition sur le site d'EnsEMBL : `http://www.ensembl.org/` (Hubbard et al., 2009). La première version d'orthoMaM, en 2007, ne contenait des séquences que pour 12 espèces. La version de juillet 2010 inclut 36 espèces, un nombre suffisant pour étudier certains processus évolutifs comme la conversion génique biaisée.

2.2.2 OrthoMaM : Construction, contenu et fonctionnalitées

La première version d'OrthoMaM a été mise en ligne en juillet 2007 dès l'acceptation par *BMC Evolutionnary Biology* de l'article qui décrit ce travail (Ranwez et al., 2007b). La mise en place du site Web permettant d'interroger OrthoMaM a bénéficié du travail efficace d'un groupe d'étudiants du mastère bio-informatique de l'UM2 que j'ai encadré. Ces étudiants sont d'ailleurs nommément remerciés à la fin de l'article. La sixième version d'OrthoMaM qui vient d'être mise en ligne (juillet 2010), fait l'objet d'un article soumis pour le numéro annuel, spécial base de données, de la revue *Nucleic Acids Research* (Douzery et al., soumis). Avec plus d'une version par an, il est clair que le fait de maintenir OrthoMaM à jour est une tâche fortement chronophage. Cependant, OrthoMaM est devenue une ressource centrale dans les travaux de notre équipe, et sa gestion m'a permis de cerner de nouveaux verrous méthodologiques.

2.2.2.1 Identifications des marqueurs

Le projet EnsEMBL se focalise sur les données moléculaires des génomes de mammifères qui sont complètement séquencés et assemblés. Il fournit pour chaque gène une annotation riche incluant notamment les limites de ses exons et des informations sur les orthologues de ce gène. Cette dernière annotation est le résultat d'une analyse phylogénétique des différentes familles de séquences homologues. Ces familles sont construites en considérant pour chaque gène son CDS le plus long puis, en appliquant un algorithme de «clustering» sur ces CDS ; cette approche phylogénétique (Li et al., 2003) permet d'identifier l'orthologie de deux séquences de manière plus fiable que la simple comparaison directe de ces deux séquences (Chen et al., 2007a). Elle permet également d'avoir une annotation plus fine et de différencier les différents types d'orthologies (Figure 2.2).

Nous nous appuyons donc sur ces annotations pour identifier des marqueurs potentiellement utiles à l'étude de la phylogénie des mammifères placentaires. Cette identification se fait en plusieurs étapes. La première étape utilise uniquement les séquences des trois espèces suivantes : l'homme (*Homo sapiens*), la souris (*Mus musculus*) et le chien (*Canis familiaris*) dont les génomes ont été séquencés avec une profondeur importante, qui sont bien annotés et qui sont suffisamment divergents pour qu'un gène conservé chez ces trois espèces ait une forte probabilité d'être conservé chez les autres mammifères placentaires. Sur la base de ces trois espèces, nous sélectionnons alors les gènes qui sont orthologues 1 :1 pour les couples d'espèces *Homo*/*Mus*, *Homo*/*Canis* et *Mus*/*Canis*. Pour chaque gène humain ayant passé ce filtre, nous identifions ensuite, s'il existe, son gène orthologue 1 :1 chez chacune des espèces de mammifères placentaires disponibles dans EnsEMBL (soit 36 espèces au total pour la version 6 d'OrthoMaM) et nous récupérons son CDS le plus long. Puis, pour chaque exon de plus de 400 paires de bases du CDS humain, nous cherchons à déterminer l'exon qui lui est orthologue dans chacun des CDS précédemment identifiés. Pour une espèce donnée, nous considérons que l'exon orthologue est celui qui possède la séquence la plus similaire à l'exon humain. Cependant, si cette similarité

ne dépasse pas 50% alors aucun exon n'est retenu pour cette espèce. Cette manière basique d'identifier les exons orthologues fonctionne de manière satisfaisante car elle bénéficie de l'annotation de l'orthologie des CDS faite par EnsEMBL. La majorité des cas problématiques, notamment ceux liés aux duplications de gènes, sont donc déjà gérés au niveau des CDS.

La première version d'OrthoMaM ne contenait que des marqueurs potentiels de type exon. L'idée première d'OrthoMaM était en effet d'identifier des marqueurs pouvant être séquencés facilement avec un séquenceur capilaire. Or, cette technologie ne permettait pas de séquencer de longs fragments génomiques. Un CDS constitué d'exons courts séparés par des introns longs était donc très difficile à séquencer. De plus, on passait déjà d'une dizaine de marqueurs classiquement utilisés en phylogénie à plusieurs milliers de marqueurs potentiels, ce qui semblait à la fois prometteur et suffisant. Cependant, au fur et à mesure de l'augmentation du nombre d'espèces disponibles, il devenait de plus en plus tentant de faire des analyses sur la base de ces seules espèces (sans recourir à un séquençage supplémentaire). La démocratisation des nouvelles technologies de séquençage, qui permet désormais de séquencer à moindre coût une grande partie de l'ensemble des ARNm d'une espèce (son *transcriptome*), nous a définitivement convaincu d'inclure les données de type CDS dans la version 4 d'OrthoMaM (juillet 2008). Les exons gardent cependant tout leur intérêt pour certaines analyses, notamment lorsqu'il s'agit de prendre en compte des phénomènes très localisés sur les chromosomes tels que la conversion génique biaisée. Pour ce type d'analyse les CDS sont moins adaptés car ils sont constitués de fragments issus de localisations différentes (à cause des introns). Le fait de ne considérer que les orthologues 1 :1 était aussi lié aux techniques de séquençage existantes. En cas de copies multiples trop semblables, il était, en effet difficile de s'assurer que l'on séquençait la bonne. La présence de copies multiples similaires continue de poser des problèmes pour l'acquisition des séquences (elles sont difficiles à assembler) et complique la détermination fiable de l'orthologie qui est capitale en phylogénie. Pour toutes ces raisons, nous continuons donc à privilégier l'orthologie 1 :1.

La version actuelle (version 6) contient 6 056 marqueurs potentiels de type exon et 12 777 de type CDS. Un marqueur n'est présent dans la base que s'il contient des séquences d'au moins 4 espèces. Etant donné l'existence de filtres supplémentaires pour déterminer l'orthologie des exons (>400pb, >50% de similarité), les marqueurs de type CDS sont plus nombreux et recouvrent généralement plus d'espèces que ceux de type exon (Figure 2.4).

2.2.2.2 Principaux descripteurs

Pour traduire *in silico* une séquence d'ADN en acides aminés (AA) à l'aide du code génétique, il faut s'assurer que l'on a bien la séquence codante (et non celle du brin complémentaire), qu'on la lit dans le bon sens (i.e. 5' vers 3') et que l'on commence la traduction à partir d'un nucléotide qui est en première position d'un codon (i.e. qu'on utilise le bon cadre de lecture). Pour simplifier la traduction et les analyses ultérieures, nous récupérons chaque séquence dans le sens 5' vers 3' en

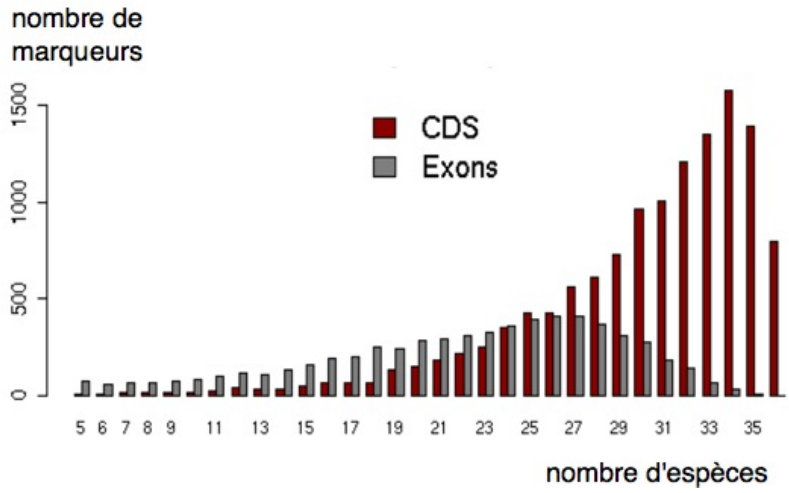

FIGURE 2.4 – **Distribution du nombre d'espèces par marqueurs dans OrthoMaM v6.**

tronquant éventuellement ses extrémités, d'un ou deux nucléotide(s), pour qu'elle commence sur une première base d'un codon et finisse une troisième.

Les séquences obtenues sont ensuite alignées. L'alignement des séquences nucléotidique (NT) est dérivé de celui de leurs traductions en AA (ce processus est décrit en détail dans la section 2.3). Puis, l'alignement nucléotidique est utilisé pour inférer la phylogénie de ce marqueur selon le principe du maximum de vraisemblance et un modèle d'évolution des séquences réaliste : le modèle GTR (General Time Reversible) couplé à l'utilisation d'une loi gamma pour modéliser l'hétérogénéité de la vitesse d'évolution des différents sites. Le principe de l'inférence par maximum de vraisemblance et les différents modèles d'évolutions classiquement utilisés sont décrits en détail dans le premier chapitre de ma thèse (Ranwez, 2002). Pour chaque descripteur, nous fournissons une fiche détaillée (Figure 2.5) qui commence par des informations générales concernant le gène dont il est issu : le symbole officiel de ce gène, son descriptif et les concepts de la *Gene Ontologie* qui l'annotent. Plusieurs autres descripteurs, estimés sur la base des alignements de séquences, sont ensuite fournis. Notamment le nombre d'espèces présentes, le nombre de sites de l'alignement NT et son pourcentage de sites variables, la composition en base des séquences et la déviation de cette composition par rapport à une distribution homogène (ou RCV pour Relative Composition Variability). Le modèle d'évolution le plus adapté pour l'analyse de ces données, obtenu par modeltest (Posada, 2003) est également fournit.

Le code génétique est fait de telle sorte que la troisième position du codon peut souvent être modifiée sans que cela ne change l'acide aminé qu'il code. Par exemple, la proline peut, indifféremment, être obtenue à partir des quatre codons commençant par "CC" (CCA, CCC etc.). La pression sélective est donc généralement moins forte en troisième position et son taux de GC (ou *GC3*) est fortement corrélé à celui de la région génomique environnante. Nous fournissons donc le GC3 de chaque marqueur

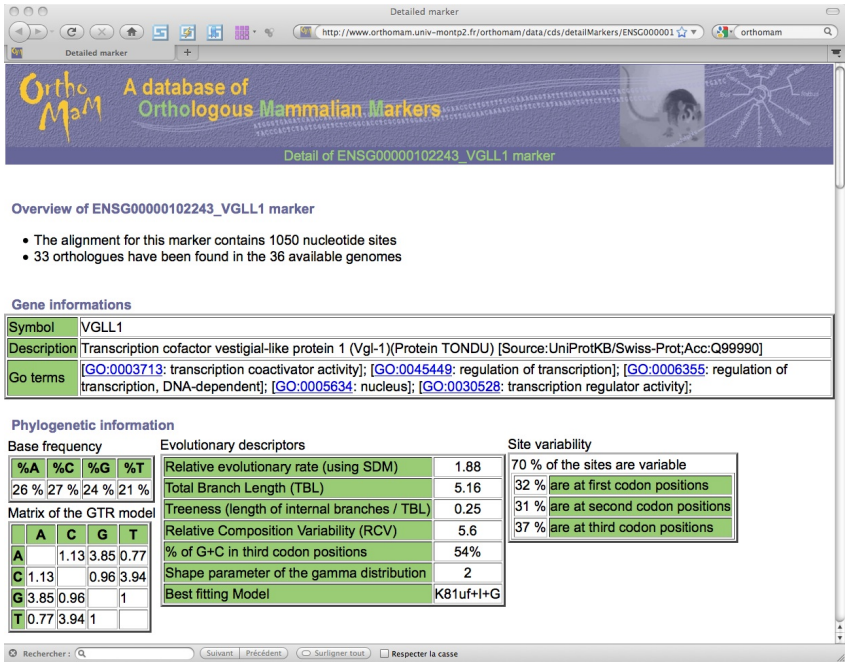

FIGURE 2.5 – **Début de la fiche détaillée d'un marqueur d'OrthoMaM v6.**

ainsi que des informations sur la manière dont la variabilité des sites se répartit sur les trois positions de ses codons

Nous fournissons également des descripteurs qui découlent de l'analyse phylogénétique du marqueur. Cela inclut notamment les paramètres du modèle d'évolution (optimisés lors de la recherche par maximum de vraisemblance) et la vitesse évolutive de ce marqueur que nous estimons (relativement à celle des autres marqueurs) à l'aide du logiciel SDM (Criscuolo et al., 2006). Enfin, l'arbre lui-même est évidemment fourni.

2.2.2.3 Quelques fonctionnalités du serveur web

Notre serveur web permet d'interroger OrthoMaM de manière simple en utilisant un formulaire (Figure 2.6). Il est notamment possible de rechercher les marqueurs en fonction de leur vitesse d'évolution (suivant que l'on souhaite les utiliser pour résoudre des phylogénies récentes ou profondes) ou du nombre de sites de leurs alignements nucléotidiques. Il est également possible de restreindre la recherche aux marqueurs contenant certaines espèces d'intérêts.

La liste des marqueurs répondant à ces critères est ensuite affichée de manière synthétique sous forme de tableau. Cette page permet de récupérer les données (alignements, arbres ou fiches complètes) d'un grand nombre de marqueurs d'intérêts en quelques clics. Enfin, si l'on sélectionne un de ces marqueurs, on accède à sa fiche détaillée (Figure 2.5).

En complément de cette interface, nous travaillons actuellement à la mise en place d'un "web service" (`http://www.orthomam.univ-montp2.fr/orthomam/`

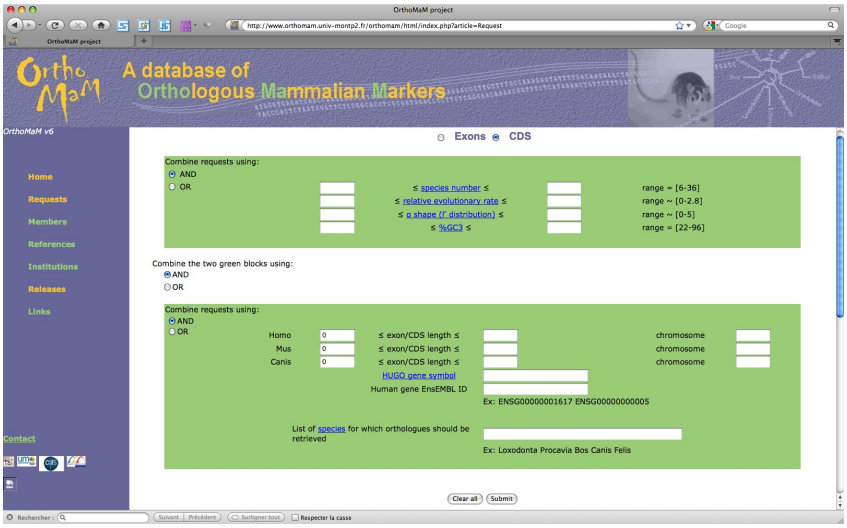

FIGURE 2.6 – **Formulaire de requête d'OrthoMaM v6.**

`html/OrthomamWS1.html`) permettant d'interroger et de récupérer des données d'OrthoMaM par l'intermédiaire de programmes.

2.2.3 Résultats biologiques s'appuyant sur OrthoMaM

L'un des intérêts majeurs d'OrthoMaM est la possibilité d'en extraire des jeux de données pertinents pour des analyses de génomique évolutive à l'échelle des mammifères. Un nombre important de questions biologiques nécessitent l'analyse d'alignements de séquences orthologues incluant de nombreuses espèces. C'est notamment le cas de nos travaux sur l'évolution du contenu en GC des génomes de mammifères. Cette section résume les résultats de ces travaux qui ont fait l'objet de deux publications, l'une dans la revue *Trends in Genetics* (Galtier et al., 2009) et l'autre dans la revue *Genome Research* (Romiguier et al., 2010). Cette seconde publication est le fruit du travail mené par Jonathan Romiguier durant son stage de Master 2 et sa première année de thèse. Je faisais partie des encadrants de Jonathan lors de son Master 2 et je co-encadre actuellement, avec Nicolas Galtier, sa thèse. La structure en isochore des génomes mammaliens à été étudié, essentiellement chez les primates et les rongeurs, en utilisant le GC3 des gènes comme approximation du GC génomique local (Mouchiroud et al., 1988; Robinson et al., 1997; Galtier and Mouchiroud, 1998). L'intérêt d'utiliser les troisièmes positions des codons est lié au fait que leur taux de GC est similaire à celui des séquences non codantes voisines (car elles subissent peu de pression sélective directe) mais qu'il est beaucoup plus facile d'identifier leurs nucléotides orthologues (grâce à la conservation des positions 1 et 2 des codons). De nombreux travaux concordaient sur le fait que la structure en isochore des mammifères tendait à s'éroder au cours de l'évolution (Duret et al., 2002; Belle et al., 2004; Li et al., 2008) et que cette érosion était plus rapide chez les rongeurs que chez les primates. Cette vision de l'évolution des isochores se basait

cependant sur des jeux de données de petite taille contenant typiquement de 50 à 500 gènes et de 4 à 8 espèces appartenant essentiellement à un sous-groupe des mammifères : les *Euarchontoglires* (primates + rongeurs + lagomorphes [1]). L'utilisation d'OrthoMaM a permis d'étudier l'évolution des isochores à une échelle plus large (en incluant beaucoup plus de gènes et d'espèces) qui a complètement remis en cause la vision de l'évolution des isochores (Romiguier et al., 2010).

2.2.3.1 Diversité de l'évolution de la structure en isochore

Pour cette analyse, nous avons utilisé les marqueurs CDS qui contenaient une séquence de chacun des mammifères présents dans la version 5 d'OrthoMaM. Nous avons ainsi obtenu un jeu de données constitué de 1 138 gènes et de 33 espèces couvrant les 4 groupes majeurs de mammifères et incluant des marsupiaux et des monotrèmes (utilisés comme groupe externe). La présence d'espèces extérieures au groupe étudié est essentielle pour pouvoir orienter les changements observés. Nous avons ensuite inféré le GC3 des séquences ancestrales de chaque CDS en utilisant un modèle de Markov non homogène pour simuler le processus d'évolution des séquences. Le fait d'étudier un large panel d'espèces nous a permis de mettre en évidence des profils d'évolution du GC3 très différents d'un groupe taxonomique à l'autre. Cette hétérogénéité est particulièrement flagrante chez le gène KIDINS220 (Figure 2.7) dont le GC3 a fortement varié depuis la divergence des thériens (les mammifères qui portent leurs petits par opposition à ceux qui pondent des œufs). Le GC3 a énormément augmenté pour une espèce de chauve-souris (*microbat*) et pour le cochon d'inde (*guinea pig*) mais a diminué chez l'oppossum et le tatou (*armadillo*). Une analyse globale de ces 1 138 gènes nous a permis de mettre en évidence des évolutions inattendues de la structure en isochores de nombreuses espèces qui n'avaient jamais été incluses dans ce type d'analyse. Comparé au génome ancestral estimé, celui de la musaraigne (*shrew*) a, par exemple, subi une forte augmentation de la valeur moyenne et de l'hétérogénéité entre gènes (variance) de son GC3. C'est également le cas du génome de la chauve-souris (*bat*) et du tenrec. Pour mieux illustrer la diversité de l'évolution des isochores, nous avons comparé les GC3 de différents couples d'espèces. Chaque couple est constitué de l'homme (dont la structure en isochore est très proche de celle du génome ancestral) et d'une autre espèce : la souris (*mouse*), la chauve-souris (*bat*), le paresseux (*sloth*) et l'ornithorynque (*platipus*). Pour chaque comparaison, nous utilisons l'ensemble des marqueurs d'OrthoMaM contenant ces deux espèces. Le nombre de gènes utilisés varie donc suivant l'espèce considérée. Chacune de ces comparaisons est résumée par un nuage de points dans lequel un gène est représenté par un point dont l'abscisse figure le GC3 humain et l'ordonnée celui de la seconde espèce. Pour chacun de ces nuages de points nous avons calculé sa droite de régression linéaire, la pente de cette droite et la qualité de cet ajustement linéaire à l'aide du r^2 (Figure 2.8). La comparaison homme/souris, ici basé sur 9 823 gènes, est consistante avec les analyses antérieures (Mouchiroud et al., 1988). La pente de la droite de régression, nettement plus faible que 1, indique une érosion

1. C'et l'ordre animal qui contient les lapins. Et oui, les lapins ne sont pas des rongeurs...

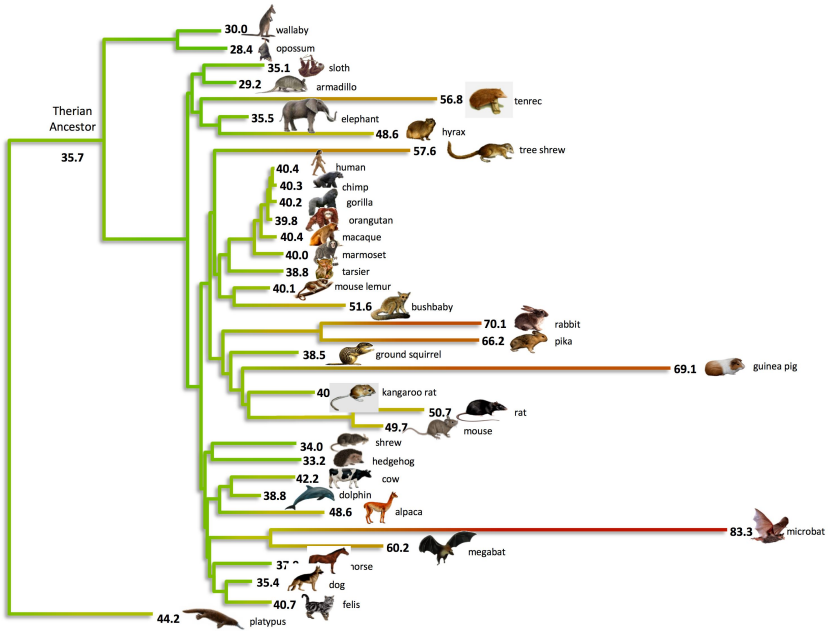

FIGURE 2.7 – **Evolution du taux de GC3 du gène KIDINS220 chez les mammifères.** *La couleur des branches repésente le taux de GC3 ancestral estimé dans ces branches. Leur longueur représente la variation du taux de GC3 entre leurs extrémitées.*

de la structure en isochore plus rapide chez les rongeurs que chez l'homme (les résultats sont similaires avec le rat). Par contre le GC3 moyen de la chauve-souris est nettement supérieur à celui de l'homme. Certains gènes de la chauve-souris (dans le coin en haut à gauche) montrent même une augmentation spectaculaire de leur GC3. S'il s'agissait d'erreurs dans les données, telles que des contaminations par des paralogues, on s'attendrait à observer également des points en bas à droite. En effet, il n'y a aucune raison pour que les paralogues de la chauve-souris soient systématiquement plus riches en GC3 que le gènes humain auquel on les compare. L'absence de points en bas à droite réfute l'hypothèse d'une contamination due à des paralogues et confirme qu'il s'agit bien d'un signal biologique. On observe également ce phénomène lorsque l'on compare l'homme à plusieurs autres espèces (dont la musaraigne et le tenrec). La comparaison avec le paresseux fournit une image encore différente de l'évolution du GC3. Le GC3 des gènes du paresseux est en effet très proche de celui de l'homme et cela bien que leur ancêtre commun soit beaucoup plus ancien que celui de l'homme et de la souris. Il semble que leur structuration en isochore ait très peu évoluée par rapport à celle de l'ancêtre commun des mammifères. A l'opposé, il semble que la structure en isochore de l'ornithorynque ait été profondément remaniée depuis la divergence des mammifères.

La diversité de ces quatres comparaisons reflète la complexité de l'évolution de la structuration en isochores des gènomes de mammifères qui n'a pu être découverte qu'à travers une analyse à grande échelle rendue possible par l'utilisation d'OrthoMaM.

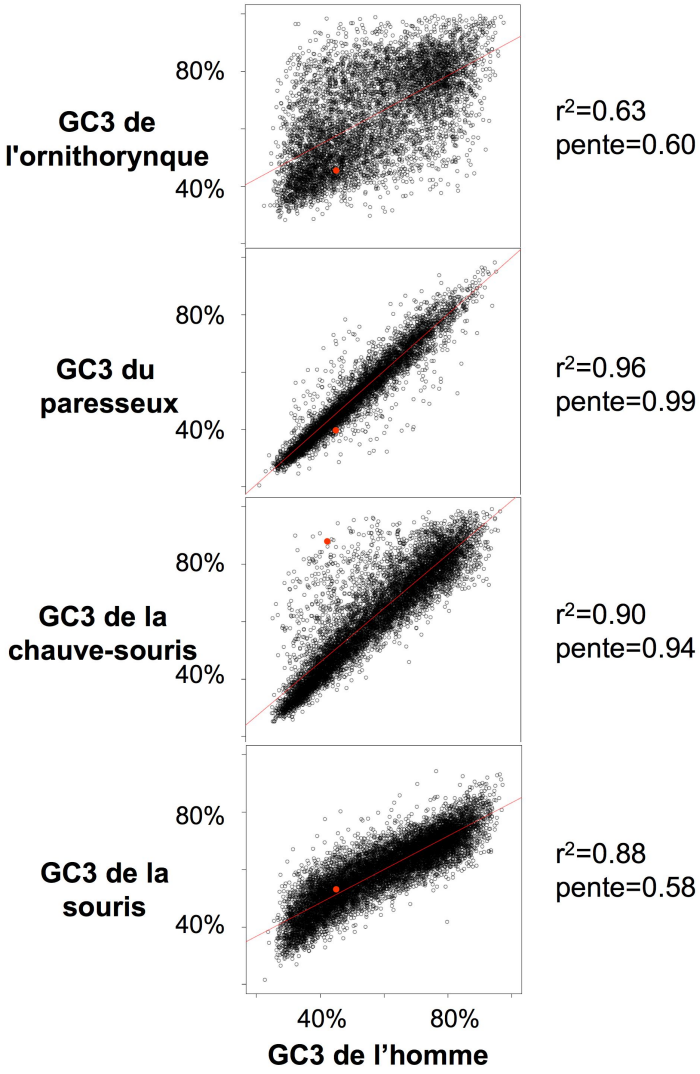

FIGURE 2.8 – **Comparaison du GC3 de gènes orthologues pour différents couples d'espèces.** *Le GC3 des CDS de l'homme est comparé avec celui de l'ornythorynque, du paresseux de la chauve-souris et de la souris. Chaque gène est représenté par un point dont l'abscisse figure le GC3 humain et l'ordonnée celui de la seconde espèce. La droite de régression linéaire est représentée pour chacun de ces nuages de points. La pente de cette droite et la qualité de l'ajustement linéaire (r^2) sont également indiqués. Les points rouges correspondent au gène KIDINS220 dont nous avons détaillé l'évolution du GC3 (Figure 2.7).*

2.2.3.2 Confirmation du rôle de la conversion génique biaisée dans la structuration en isochore

Comme évoqué ci-dessus, il existe chez la chauve-souris des gènes dont le GC3 augmente de manière brutale et complètement atypique (par rapport à l'évolution des autres gènes). L'existence de ces phénomènes d'augmentation rapide du GC3 conforte l'hypothèse du rôle de la conversion génique biaisée dans l'évolution du GC des mammifères. En effet, nous avons vu que cette dernière agissait en biaisant les réparations nécessaires suite à un évènement de recombinaison (Figure 2.3). Hors, ces évènements se produisent essentiellement à des endroits précis du génome (appelés points chauds de recombinaison) dont la position change rapidement (Winckler et al., 2005). Ainsi lorsqu'un point chaud de recombinaison apparaît à proximité d'un gène, ce dernier subit de nombreux évènements de recombinaison et de réparation biaisée. Il fixe donc rapidement un grand nombre de mutations vers GC (comme toute la région génomique voisine). Puis, quand peu de temps après, ce point chaud de recombinaison disparaît, le GC de ce gène diminue lentement, car les mutations vers AT sont en moyenne légèrement plus fréquentes que celles vers GC (Webster and Smith, 2004).

2.2.3.3 La conversion génique biaisée est un phénomène égoïste

Une des prédictions théoriques liées à ces évènements de conversion génique biaisée est qu'ils peuvent également augmenter fortement les substitutions au niveau des acides aminés. En effet durant ces épisodes, des mutations AT→GC légèrement délétères, et qui seraient donc normalement éliminées par la sélection négative (purifying selection) peuvent être fixées à cause du biais favorisant les allèles GC (Galtier and Duret, 2007). Pour vérifier cette hypothèse nous avons utilisé OrthoMaM pour chercher dans les génomes de primates des accélérations subites du taux d'évolution au niveau des acides aminés. Nous avons ainsi analysé les 12 425 alignements d'exons incluant simultanément une séquence de l'homme, du chimpanzé, de l'orang-outan, du macaque et de l'une au moins des deux espèces suivantes le lémur et le galago (*bushbaby*) pour servir de groupe externe. Nous avons ensuite construit un arbre de référence en utilisant une approche de type super-matrice basée sur la concaténation des alignements d'AA de 1 000 exons choisis aléatoirement. La longueur des branches de l'arbre ainsi obtenu représente la quantité moyenne d'évolution (en AA) entre deux évènements de spéciation. Nous avons ensuite estimé les longueurs de ces branches à partir de chacun des 12 425 exons. Chaque fois qu'une branche était anormalement plus longue pour un exon que pour l'arbre de référence nous avons considéré qu'il s'agissait d'un épisode d'accélération de l'évolution des AA. Nous avons ensuite analysé les 622 événements ainsi détectés et mis en évidence une augmentation significative des mutations de type AT→GC au cours de ces épisodes et cela à la fois pour des mutations synonymes (même traduction du codon contenant cette mutation) et non synonymes. Cette utilisation d'OrthoMaM nous a ainsi permis de confirmer l'impact de la conversion génique biaisée sur les séquences d'acides aminés.

2.3 Alignement multiple de séquences codantes

2.3.1 Motivations

Dès ma première année en tant que Maître de Conférences, j'ai collaboré avec Emmanuel Douzery pour monter une nouvelle série de travaux pratiques (TP) autour de la phylogénie moléculaire. L'un de ces TP portait sur l'alignement de séquences. Les étudiants devaient récupérer les séquences nucléotidiques d'un exon du gène BRCA1 pour plusieurs mammifères, aligner ces séquences et répondre à des questions liées à l'observation de cet alignement. Ces questions les amenaient à constater que dans la majorité des cas les *indels* (i.e. les insertions ou les délétions) concernent un nombre de nucléotides qui est multiple de trois ; puis à comprendre que cela est lié à la structuration en codon des nucléotides. En effet, insérer 6 nucléotides revient à insérer 2 AA alors qu'insérer 5 nucléotides change le cadre de lecture et modifie en profondeur toute la séquence d'AA qui suit cette insertion (ce qui a de grandes chances de rendre la séquence non fonctionnelle). Les étudiants devaient ensuite localiser dans l'alignement des cas où les *gaps* (les successions de "–") ne sont pas multiples de 3 et proposer une explication possible de ce phénomène. Dans la plupart des cas, ces gaps fonctionnent par paires. Ils représentent, par exemple, une insertion de 2 nucléotides suivie peut après d'une insertion de 7 qui permet de récupérer le bon cadre de lecture (2+7 donne un multiple de 3). Un alignement alternatif, avec une insertion de 3 nucléotides puis une de 6, est sans doute meilleur, mais les logiciels d'alignement n'exploitent pas la structuration en codon. Pour eux ces deux alignements sont aussi valides l'un que l'autre. C'est l'une des raison qui conduit les biologistes à retoucher manuellement les alignements produits par ces logiciels. Cela n'invalide nullement l'utilité de ces outils pour aligner des séquences codantes car il est nettement plus simple de retoucher un alignement que de le faire en partant des séquences brutes. L'une des conclusions du TP était que cet exemple illustre parfaitement les forces et les limites des outils bio-informatiques et la nécessité de garder un œil critique sur les résultats qu'ils nous fournissent.

A force de répéter que les logiciels d'alignement de séquences nucléotidiques ne prennent pas en compte la structuration en codon, j'ai fini par me demander comment il serait possible de le faire. Une solution permettant de contourner le problème consiste à utiliser une *stratégie en trois étapes* où 1) les séquences nucléotidiques sont traduites en AA, 2) ces séquences d'AA sont alignées et 3) l'alignement nucléotidique est déduit de cet alignement d'AA (Suyama et al., 2006; Bininda-Emonds, 2005; Abascal et al., 2010). Cette approche en trois étapes, utilisée pour aligner les séquences d'OrthoMaM, est cependant loin de résoudre tous les problèmes. En effet, dès qu'une séquence contient un codon stop ou un nucléotide en plus, ou en moins, plus rien ne fonctionne. Or, avec l'arrivée des nouvelles technologies de séquençage, nous avions, comme tout le monde, de plus en plus de séquences contenant des erreurs qu'il était impossible d'aligner en utilisant cette stratégie en trois phases (les contigs produits par la phase d'assemblage cf 1.2.3). Parallèlement, Frédéric Delsuc (CR dans notre équipe) s'est intéressé à l'analyse de l'évolution des dents notam-

ment au travers de l'analyse moléculaire du (pseudo-)gène AMBN qui n'est plus fonctionnel chez certaines espèces de baleines. Là encore, aucune solution d'alignement existante n'était satisfaisante. Enfin, à force d'utiliser OrthoMaM, notamment dans le cadre de projets sur la conversion génique biaisée, nous avons constaté que certaines séquences d'EnsEMBL contenaient, elles aussi, des nucléotides erronés qui faussaient les alignements. Le développement d'un outil d'alignement de séquences capable d'intégrer des évènements de changement de cadre de lecture et la présence de codon stop s'est donc petit à petit imposé comme une nécessité.

Au printemps 2009, j'ai donc proposé un stage de Master 2 sur ce sujet. Sébastien Harispe, que j'ai encadré seul, a fait un excellent travail et à la fin de son stage nous avions une solution qui donnait des alignements de bonne qualité pour deux séquences. L'extension à l'alignement multiple était prometteuse, mais nécessitait d'être améliorée. Un an plus tard, nous avons maintenant un outil pleinement fonctionnel qui est régulièrement utilisé au sein de l'équipe. La génèse de ce travail reflète parfaitement la complémentarité de l'enseignement et de la recherche ainsi que les interactions qui peuvent exister entre des projets appliqués (OrthoMaM), d'analyse de données (sur la conversion génique biaisée) et de recherche algorithmique. Ce chapitre est un résumé de ces travaux, qui font l'objet d'une publication actuellement soumise à *Molecular Biology and Evolution* et dont Sébastien est co-auteur (Ranwez et al., soumis). Un serveur Web (`http://mbb.univ-montp2.fr/macse`) permet d'utiliser, et de télécharger le programme qui découle de ces travaux (il sera rendu public dès l'acceptation de l'article).

2.3.2 Intérêts de prendre en compte la traduction en AA

Une séquence codante peut être considérée au niveau nucléotidique (NT) ou au niveau acide aminé (AA) ces deux niveaux étant les deux faces d'une même pièce. Du fait de la redondance du code génétique, différents codons peuvent représenter le même AA. La séquence nucléotidique est donc moins conservée (car moins contrainte) mais plus informative que sa traduction en AA. Un alignement n'utilisant que le niveau nucléotidique perd l'avantage lié à la plus forte conservation des AA et ne prend pas en compte le fait que les indels impliquent généralement un nombre de nucléotides qui est multiple de 3 (Figure 2.9). A l'inverse, la stratégie en trois étapes, décrite plus haut, en alignant les séquences uniquement sur la base de leurs traduction en AA s'interdit d'introduire des changements de cadre de lecture ou des codons stops. Ces évènements ne sont pourtant pas si rares pour certains types de séquences. Les changements de cadre de lecture dus à des erreurs de séquençage sont notamment fréquents dans les séquences obtenues par les nouvelles technologies de séquençage (Margulies et al., 2005). Ces séquences peuvent également contenir des codons stop lorsqu'une base est mal lue. Des changements de cadre de lecture et des codons stops "réels" sont également fréquents chez des pseudogènes et sont même parfois observés dans des séquences fonctionnelles. En effet, il arrive que la "machinerie moléculaire" tolère des changements de cadre de lecture (Russell and Beckenbach, 2008) ou la présence de codon stop (Williams et al., 2004).

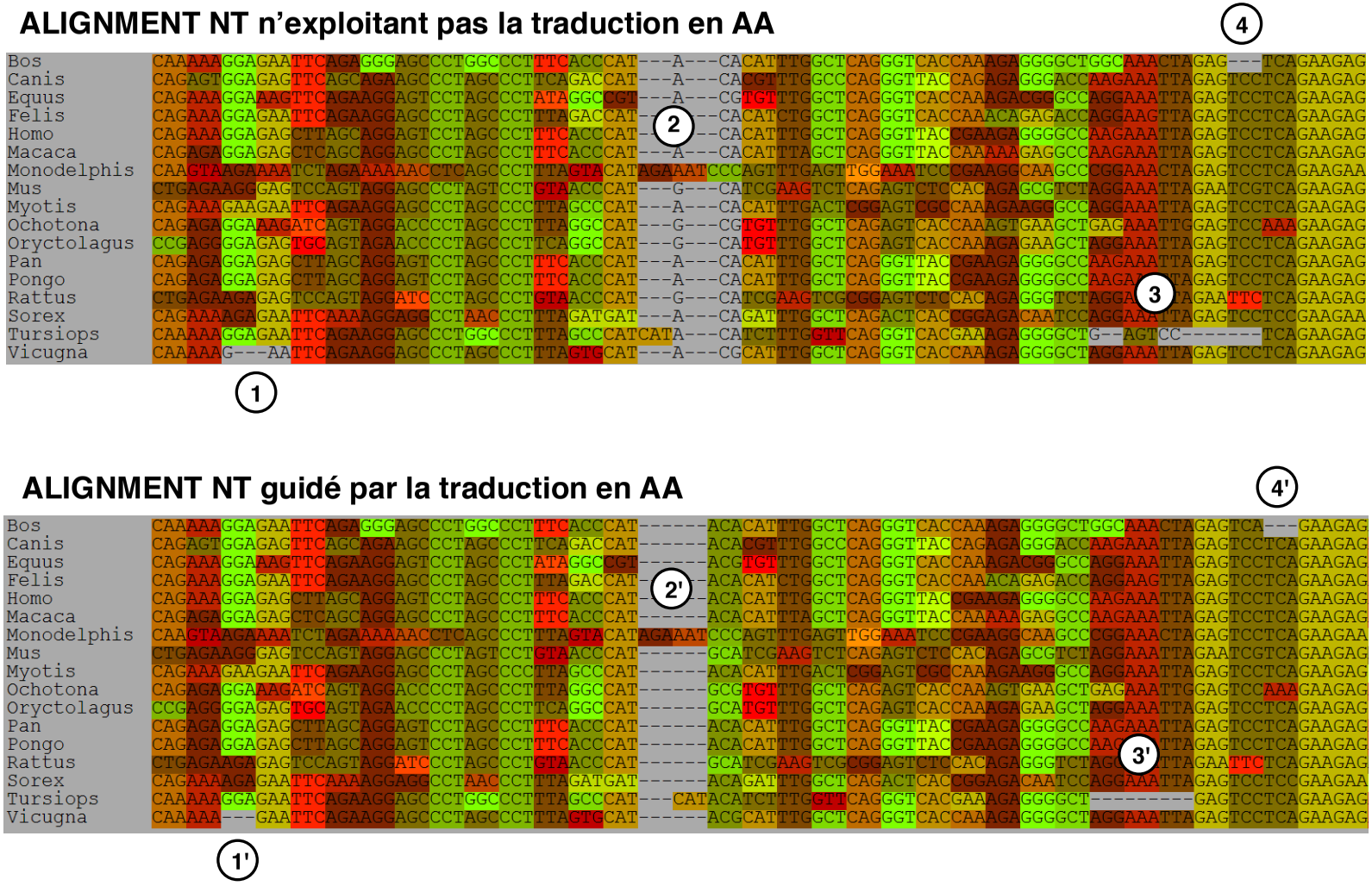

FIGURE 2.9 – **Différents alignements d'ADN codants obtenus en prenant (ou non) en compte leur traduction en AA.** *Cas 1 et 1' : la délétion de 3 nucléotides chez* Vicugna *ne respecte pas le cadre de lecture lorsque la traduction en AA n'est pas prise en compte. Cas 2 et 2' : là encore le cadre de lecture n'est pas respecté pour l'insertion de 6 nucléotides chez* Monodelphis. *Cas 3 et 3' : les délétions de 2 puis de 7 nucléotides chez* Tursiops *se compensent, mais ne sont pas multiples de 3 individuellement. Cas 4 et 4' : la délétion de 3 nucléotides de* Bos *est correctement positionné par l'alignement n'utilisant pas la traduction en AA, mais une position alternative est proposée lorsqu'on utilise un alignement guidé par la traduction en AA.*

Etant plus informatives, les séquences NT devraient permettre d'obtenir des alignements au moins aussi bons que ceux n'utilisant que leur traduction en AA. En effet, à partir des séquences NT, il est possible de prendre en compte les conservations qui ne sont visibles qu'au niveau AA (en traduisant les codons à la volé) ainsi que de favoriser la préservation du cadre de lecture (tout en prenant en compte la possibilité qu'il puisse parfois changer). L'alignement de séquences étant une étape clef de la majorité des analyses moléculaires, il existe de nombreux travaux sur le sujet ayant donné lieu à une multitude de logiciels très utilisés en bio-informatique tels que CLUSTAL (Higgins et al., 1992), T-COFFEE (Notredame et al., 2000), MUSCLE (Edgar, 2004b), MAFFT Katoh et al. (2005), PRANK (Löytynoja and Goldman, 2008) et FSA (Bradley et al., 2009).

Paradoxalement, il existe assez peu de travaux concernant l'alignement de séquences NT exploitant leur traduction en AA. Une solution a été proposée par Hein (1994) sur la base d'un modèle général ADN/protéine où le coût global d'un alignement est une combinaison des coûts des deux alignements correspondants (NT et AA). Après avoir décrit ce modèle général, Hein (1994) se concentre sur une instance particulière où : i) le coût global est simplement la somme des deux coûts, ii) les changements de cadre de lecture ne sont pas autorisés et iii) le coût d'un gap est linéaire en fonction de sa taille. Il décrit un algorithme permettant de trouver l'alignement optimal de deux séquences suivant ce modèle simplifié. Pour aligner une séquence de n nucléotides avec une autre de m nucléotides, cet algorithme requiert (pour des valeurs de n et de m suffisamment grandes) un nombre d'opérations élémentaires proportionnel à n^2m^2. On dit d'un tel algorithme qu'il a une *complexité en temps* de $O(n^2m^2)$. Cette solution a ensuite été étendue, par Pedersen et al. (1998), au cas où le coût d'un gap est représenté par une fonction affine, tout en réduisant la complexité en temps à $O(nm)$. Biologiquement, il est beaucoup plus réaliste d'utiliser une fonction affine pour calculer le coût d'un gap, cela permet de modéliser le fait qu'une insertion (ou une délétion) est un évènement évolutif relativement rare et qu'il est plus rare d'observer deux insertions de 3 nucléotides qu'une seule de 6. Cette solution semblait donc prometteuse, d'autant plus qu'elle a la même complexité asymptotique que les algorithmes classiques permettant d'aligner deux séquences NT sans prendre en compte leur traduction en AA. Cependant, le facteur de proportionnalité, qui n'est généralement pas un problème, est dans ce cas de l'ordre de 400. Les auteurs reconnaissent eux mêmes que ce facteur, masqué par la notation O, peut être une limitation de leur solution (Pedersen et al., 1998). Par ailleurs, même si la somme des coûts des deux alignements (NT et AA) semble une manière intuitive de combiner ces deux niveaux d'information elle est difficile à justifier par rapport à d'autres combinaisons possibles (Pedersen et al., 1998). De plus, les coûts des mutations entre AA prennent déjà partiellement en compte les mutations nucléotidiques sous-jacentes. Il nous semble donc préférable de ne prendre en compte que le coût de l'alignement des AA. Mais l'inconvénient majeur de leur solution est qu'elle n'autorise pas les changements de cadre de lecture.

Une approche alternative, proposée par Stocsits et al. (2005) consiste à évaluer le coût global de l'alignement en utilisant une somme pondérée des quatre coûts

suivants : celui de l'alignement NT plus ceux des trois alignements obtenus en le traduisant suivant les trois cadres de lecture possibles. Dans ce modèle, aucun coût particulier n'est donné aux indels induisant des changements de cadre de lecture ; ils sont supposés être pénalisés par les mésappariements qu'ils engendrent au niveau des AA. Le fait de prendre en compte simultanément les trois cadres de lecture possibles peut paraître surprenant, mais cela est dû au fait que cette solution a été développée pour des séquences virales qui ont la propriété d'exploiter simultanément plusieurs cadres de lecture. Elles utilisent ainsi la même portion de séquence pour coder des protéines différentes ce qui permet d'avoir un génome extrêmement compact et plus rapide à dupliquer pour proliférer (Stocsits et al., 2005).

Dans un contexte différent, Guan and Uberbacher (1996) ont proposé un algorithme permettant de détecter les erreurs de séquençage qui induisent des changements de cadre de lecture apparents. Pour ce faire, ils comparent la séquence nucléotidique codante nouvellement séquencée avec une séquence d'AA homologue obtenue en interrogeant des banques de données publiques. L'algorithme proposé pour cette comparaison généralise l'algorithme classique permettant de comparer deux séquences (Needleman and Wunsch, 1970) de manière à prendre en compte les trois cadres de lecture de la nouvelle séquence NT. Cet algorithme fournit une solution élégante pour évaluer la proximité entre les deux séquences, mais l'alignement sous-jacent ne peut pas être représenté sous la forme matricielle classique, une étape pourtant nécessaire dans le cas d'une généralisation à l'alignement multiple.

La solution que nous proposons peut être vue comme une extension de l'approche ci-dessus où les deux séquences sont nucléotidiques et peuvent potentiellement subir des changements de cadre de lecture et contenir des codons stops. De plus, notre modèle correspond à une représentation matricielle classique de l'alignement et l'algorithme que nous proposons est donc généralisable au cas de l'alignement multiple. Le programme MACSE (Multiple Alignment of Coding Sequences) est une implémentation de cette généralisation. Nous avons utilisé les alignements de références de Carroll et al. (2007) pour vérifier que, même en l'absence de changement de cadre de lecture ou de codon stop, MACSE produisait des alignements d'une qualité comparable à celle de l'approche en trois étapes. Cependant, le véritable intérêt de MACSE réside dans sa capacité à aligner correctement des séquences contenant potentiellement de tels évènements et pour lesquelles il n'existait, jusque-là, aucun outil adapté.

2.3.3 Principes algorithmiques de MACSE

2.3.3.1 Définir le coût d'un alignement de deux séquences codantes

L'alignement de deux séquences S_1 et S_2 peut être vu comme la représentation d'une transformation permettant de changer S_1 en S_2. Par exemple, l'alignement de la Figure 2.10 décrit une manière de transformer S_1 =ISEMVMII en S_2 =ISMIVN à l'aide des opérations suivantes : i) supprimer le E, ii) insérer un I après le premier M, iii) changer le M en N et iv, v) supprimer les deux I finaux. L'alignement est

donc un problème très proche (Sellers, 1974) de celui, plus connu en informatique, qui consiste à trouver la "distance d'édition" ou "distance de Levenshtein" entre deux mots (Levenshtein, 1966).

S_1	I	S	E	M	_	V	M	I	I
S_2	I	S	_	M	I	V	N	_	_

FIGURE 2.10 – **Exemple d'alignement de deux séquences.**

Une fois qu'un coût a été associé à chaque transformation élémentaire (changer une lettre en une autre, insérer/supprimer une ou plusieurs lettres), on peut définir le coût d'un alignement comme la somme des coûts élémentaires du processus de transformation sous-jacent. Un alignement optimal est l'un de ceux ayant le coût global de transformation minimal. Pour obtenir un alignement biologiquement satisfaisant, il est capital d'apporter un soin particulier au choix des différents coûts élémentaires. Le coût associé à la transformation d'un acide aminé X en Y, noté $\sigma(X, Y)$ dans la suite, dépend de la similitude de leurs propriétés physico-chimiques. Les matrices PAM (Dayhoff et al., 1978) et Blosum (Henikoff and Henikoff, 1992) fournissent des valeurs raisonnables de $\sigma(X, Y)$. Le coût d'une insertion/délétion concernant un nombre l d'acides aminés est généralement défini par une fonction affine de la forme : $cost(gap\_open) + l * cost(" - ")$ où $cost(gap\_open)$ est une valeur élevée pénalisant l'ouverture d'un gap tandis que $cost(" - ")$ est une valeur plus faible pénalisant l'extension de ce gap.

Afin de représenter la présence de changements de cadre de lecture et de codon stop, nous introduisons deux nouveaux symboles dans l'alphabet utilisé pour représenter ces alignements. Le premier est le "!" qui, au niveau AA, représente la présence d'un codon incomplet (i.e. contenant des gaps et 1 ou 2 NT) et qui est utilisé dans les alignements NT pour représenter les gaps d'un tel codon. Le second est le caractère "*" qui est classiquement utilisé au niveau AA pour représenter la présence d'un codon stop. Comme expliqué plus haut, notre modèle s'appuie uniquement sur l'alignement des AA pour définir le coût de l'alignement de séquences NT codantes. Il nous suffit donc de définir les coûts élémentaires associés aux deux nouveaux symboles autorisés dans nos alignements d'AA pour définir complètement notre critère d'optimisation. La présence d'un changement de cadre de lecture, ou d'un codon stop, est un évènement rare, interne à la séquence qui le subit et dont la probabilité est indépendante du contenu de l'autre séquence. La présence d'un "!" (resp. "*") est donc associée à coût élevé noté $cost("!")$ (resp. $cost(" * ")$) quel que soit l'élément qui lui fait face dans l'alignement. Si un alignement contient un site où une "*" fait face à un "!" le coût de ce site sera donc $cost("!") + cost(" * ")$.

2.3.3.2 L'algorithme standart de Needleman-Wunsch

L'alignement optimal de deux séquences codantes, suivant le modèle NT/AA décrit précédemment, peut être obtenu à partir d'une variante de l'algorithme clas-

sique dit de "Needleman et Wunsch" permettant d'aligner deux séquences d'AA (Needleman and Wunsch, 1970; Sankoff, 1972; Altschul and Erickson, 1986). Cette section rappelle donc le principe de cet algorithme dans le cas simple où une fonction linéaire est utilisée pour pénaliser les gaps, i.e. le coût d'un gap de taille l est simplement $l * cost(" - ")$.

Etant donnée une séquence S, on note $lg(S)$ sa longueur et $S[i : j]$ la sous-séquence comprise entre son $i^{\text{ème}}$ et son $j^{\text{ème}}$ caractère. Notez que $S[i : i]$ est donc le $i^{\text{ème}}$ caractère de S et que par convention $S[i : j]$ est la séquence vide ("") dès que $j < 1$ ou $j < i$. La première observation clef est que l'alignement optimal de deux séquences peut se déduire facilement de l'alignement optimal de ces mêmes séquences tronquées d'au plus un caractère. Plus précisément, notons $\mathcal{A}(S_1, S_2)$ l'alignement optimal des séquences S_1 and S_2 et $cost(\mathcal{A}(S_1, S_2))$ son coût. Le coût optimal de l'alignement de deux séquences peut s'exprimer de manière récursive à l'aide de la formule suivante (tant que $i \geq 1$ et $j \geq 1$) :

$$cost(\mathcal{A}(S_1[1:i], S_2[1:j])) = min \begin{cases} cost(\mathcal{A}(S_1[1:i-1], S_2[1:j-1])) + \sigma(S_1[i], S_2[j]) \\ cost(\mathcal{A}(S_1[1:i-1], S_2[1:j])) + cost(" - ") \\ cost(\mathcal{A}(S_1[1:i], S_2[1:j-1])) + cost(" - ") \end{cases} \quad (2.1)$$

La récursion s'arrête quand au moins une des séquences est vide. Dans ce cas, l'alignement est simplement une série de i (resp. j) gaps faisant face à i (resp. j) caractères de S_1 (resp. S_2), dont le coût est $i * cost(" - ")$ (resp. $j * cost(" - ")$).

La seconde observation clef est que cette récurrence engendre un nombre exponentiel de calculs car elle nécessite de résoudre de nombreuse fois les mêmes sous-problèmes. Cependant, chacun de ces sous-problèmes correspond à l'alignement d'un préfixe de S_1 avec un préfixe de S_2, il n'y a donc que $lg(S_1) * lg(S_2)$ sous-problemes disctincts.

Algorithm 1: Calcul des coûts optimaux pour l'alignement de deux séquences (Needleman-Wunsch)

Data: Deux séquences S_1 et S_2, une fonction renvoyant le coût d'un gap ($cost(" - ")$), une autre renvoyant le coût de substitution d'un caractère "X" en "Y" ($\sigma(X, Y)$).
Result: Un tableau C tel que $C[i][j] = score(\mathcal{A}(S_1[1:i], S_2[1:j]))$.
for $i = 0$ **to** $lg(S_1)$ **do**
  for $j = 0$ **to** ; $lg(S_2)$ **do**
    if $i == 0$ *AND* $j = 0$ **then**
      C[i][j]=0 ;
    else
      if $i > 0$ *AND* $j > 0$ **then**
        $subst = \sigma(S_1[i:i], S_2[1:j])$
      else
        $subst = +\infty$

      $$C[i][j] = min \begin{cases} get\_C(i-1, j-1) + subst \\ get\_C(i-1, j) + cost(" - ") \\ get\_C(i, j-1) + cost(" - ") \end{cases}$$

return C ;

Une manière efficace de résoudre cette récurrence est donc de stocker le résultat de chacun de ces sous-problèmes plutôt que de les recalculer (*programmation dynamique*). On utilise pour cela un tableau C à deux dimensions

de taille $(lg(S_1)+1)$ x $(lg(S_2)+1)$ tel que : $C[i][j] = score(\mathcal{A}(S_1[1:i], S_2[1:j]))$. La Figure 2.11 a) donne l'exemple d'un tableau C initialisé pour deux courtes séquences et des coûts simples. La dernière case du tableau C contient le coût de l'alignement optimal de S_1 et S_2. Notez que le premier caractère d'une séquence est $S[1:1]$, tandis que, conformément à la convention classique en informatique, la première case du tableau est $C[0][0]$. Ce jeu d'indice simplifie les notations car il compense l'introduction de la ligne et de la colonne supplémentaires qui sont dans C pour initialiser la récurrence.

La valeur d'une case $C[i][j]$ peut se déduire de la formule de récurrence (2.1) qui utilise les solutions des trois sous-problèmes stockés dans les cases voisines $C[i-1][j-1]$, $C[i-1][j]$ et $C[i][j-1]$. Pour que ces données soient disponibles, il faut donc calculer les cases de C dans un ordre approprié en allant de gauche à droite et de haut en bas. La manière classique d'écrire cet algorithme commence par initier la première ligne et la première colonne de C, mais cette approche se généralise mal au cas des séquences codantes. Nous proposons donc ici une variante (Algorithm 1) qui s'appuie sur une fonction "$get\_C(i,j)$" qui renvoie la valeur de la case $C[i][j]$ si cette case existe et renvoie $+\infty$ sinon. L'avantage de cette approche est que $+\infty$ n'interfère pas avec la recherche d'un minimum. Il suffit alors juste d'initialiser $C[0][0]$ et les cases de la première ligne/colonne peuvent ensuite être calculées comme n'importe quelle autres. La complexité de cet algorithme, en temps et en espace, est $O(lg(S_1) * lg(S_2))$.

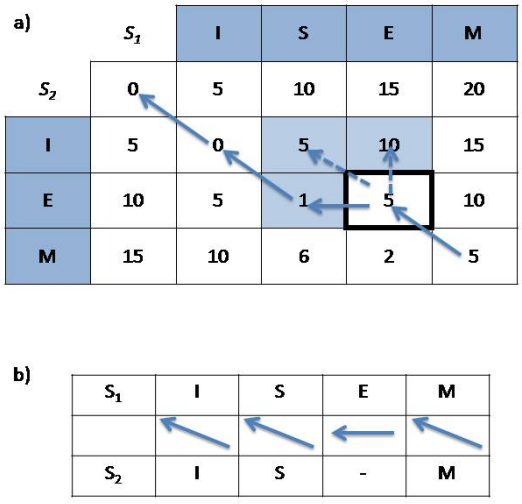

FIGURE 2.11 – **Obtention de l'alignement de deux séquences à partir du tableau C.** *Une fois le tableau C initialisé, ici avec les coûts simples suivants : $\sigma(X,Y) = 0$ si $X = Y$ et 1 sinon, cost("$-$")=5, il est possible de déduire l'alignement optimal en partant de la dernière case (en bas à droite).*

Une fois le tableau C initialisé, un alignement optimal des séquences S_1 et S_2 peut être obtenu en $O(lg(S_1) + lg(S_2))$. On part pour cela de la dernière case de C (i.e. $C[lg(S_1)][lg(S_2])$ et l'on détermine laquelle des trois cases voisines a permis d'obtenir la valeur optimale de cette dernière case. Si c'est la case de gauche, cela indique une insertion du dernier caractère de S_1 ; si c'est celle du dessus il s'agit d'une délétion de ce caractère et si c'est la case diagonale alors il s'agit d'une substitution entre les deux derniers caractères de S_1 et S_2. On se positionne alors sur la case qui vient d'être identifiée et le processus se répète jusqu'à arriver à la case $C[0, 0]$. Si, à une étape, plusieurs solutions sont possibles, l'un des mouvements optimaux est choisi de manière arbitraire (cela arrive lorsqu'il existe plusieurs alignements optimaux). Le nombre de ces choix constitue un indicateur possible de la fiabilité de l'alignement proposé (Landan and Graur, 2007). La Figure 2.11 donne un exemple de tableau C initialisé et détaille les étapes successives permettant d'obtenir l'alignement optimal correspondant.

2.3.3.3 Alignement de deux séquences codantes selon un modèle NT/AA intégrant les changements de cadre de lecture et les codons stops

Afin de définir un alignement de séquences NT qui prend en compte leurs traductions en AA, il est nécessaire d'introduire quelques notations supplémentaires permettant de faire le lien entre ces deux aspects des séquences. Dans la suite, nous notons $\pi(S)$ la traduction brute d'une séquence nucléotidique S. Cette traduction est basée sur le premier cadre de lecture, les codons stop sont traduits par des "*" (sans que cela interrompe la traduction de S) et les codons incomplets sont traduits par des "!".

Si l'on considère l'alignement de deux séquences codantes pour un modèle n'autorisant pas les changements de cadre de lecture, le tableau C_{AA} permettant d'aligner $\pi(S_1)$ et $\pi(S_2)$ peut être vu comme une version compressée du tableau C permettant d'aligner S_1 et S_2. En effet, chaque ligne (resp. colonne) de C_{AA} représente trois lignes (resp. colonnes) de C. Un alignement équivalent à celui produit à partir de C_{AA} peut donc être obtenu à partir de C en limitant les mouvements possibles à ceux correspondants à des insertions, des délétions ou des substitutions d'AA. Une telle restriction revient simplement à ne considérer que les cases $C[3i'][3j']$ de ce tableau tout en estimant leurs valeurs à l'aide de la formule suivante :

$$C[3i', 3j'])) = min \begin{cases} get\_C(3i' - 3,\ 3j' - 3) + \sigma(\pi(S_1[3i' - 3 : 3i']), \pi(S_2[3j' - 3 : 3j'])) \\ get\_C(3i' - 3,\ 3j') + cost(" - ") \\ get\_C(3i',\ 3j' - 3) + cost(" - ") \end{cases} \tag{2.2}$$

Afin de prendre en compte la possibilité de changement de cadre de lecture, nous avons généralisé cette approche en calculant toutes les valeurs $C[i][j]$ et en considérant pour chacune l'ensemble des cellules se trouvant dans le voisinage délimité par

$C[i][j]$, $C[i-3][j]$, $C[i-3][j-3]$ et $C[i][j-3]$. Ce carré 4x4 définit un voisinage de 15 cases pour $C[i][j]$ (Figure 2.12). Lors de la construction de l'alignement à partir des valeurs de C, les 15 possibilités sont considérées. Trois d'entre elles correspondent à des évènements classiques au niveau des AA, tandis que les 12 autres reflètent l'introduction d'une ou deux ruptures de cadre de lecture dans l'alignement (Figure 2.13). L'algorithme 2 détaille la manière dont le tableau C est initialisé suivant ce modèle.

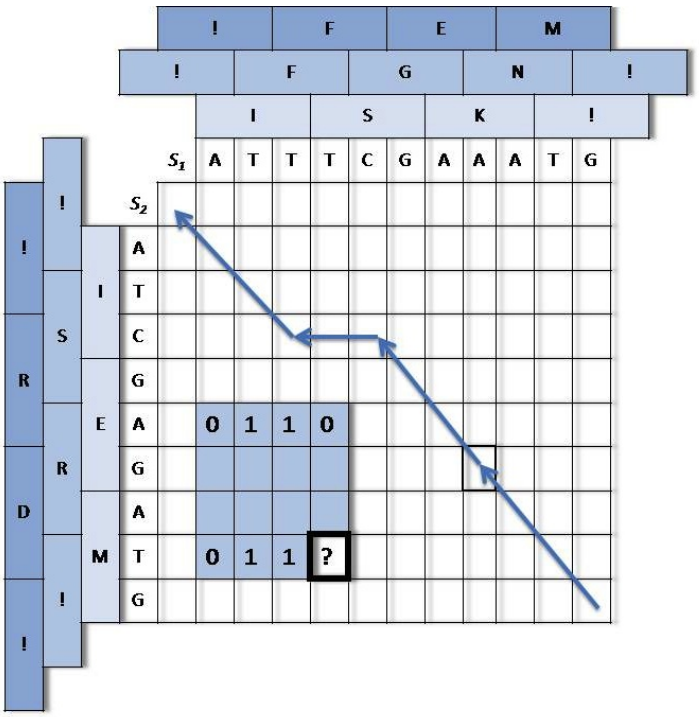

FIGURE 2.12 – **Alignement de deux séquences NT codantes.** *La traduction en AA des deux séquences à aligner – S_1=(ATTTCGAAATG) et S_2=(ATCGAGATG) – est utilisée pour détecter les codons stop et calculer les coûts de substitution. La valeur de chaque cellule est calculée à partir de 15 cellules voisines. La valeur de la cellule en gras est, par exemple, déduite en utilisant les 15 cellules colorées voisines. Parmi ces 15 cellules, les 3 marquées d'un "0" n'induisent pas de changement de cadre de lecture, tandis que les trois marquées d'un "1" induisent un changement de cadre de lecture pour S_1 mais pas pour S_2. Les relations entre ces 15 mouvements et l'alignement produit sont détaillées dans Figure 2.13*

Algorithm 2: Calcul des coûts optimaux pour l'alignement de deux séquences NT codantes.

Data: Deux séquences NT : S_1 et S_2, une méthode "*cost*" renvoyant le coût d'un changement de cadre de lecture, d'une délétion, ou d'un codon stop, une méthode σ renvoyant le coût lié à la substitution d'un AA par un autre.

Result: Un tableau C tel que $C[i][j] = score(\mathcal{A}(S_1[1:i], S_2[1:j]))$.

for $i = 0$ **to** $lg(S_1)$ **do**
  for $j = 0$ **to** $lg(S_2)$ **do**
    if $i == 0$ *AND* $j = 0$ **then**
      C[i][j]=0 ;
    else
      if $i - 3 > 0$ **then** $AA_1 = (\pi(S_1[i-3:i])$ **else** AA_1= " ?";
      if $j - 3 > 0$ **then** $AA_2 = (\pi(S_2[j-3:j])$ **else** AA_2= " ?";
      if $AA_1 ==$ " $*$ " **then** stopS_1 = cost("*") **else** stopS_1=0;
      if $AA_2 ==$ " $*$ " **then** stopS_2 = cost("*") **else** stopS_2=0;
      if $AA_1 ==$ " $*$ " *OR* $AA_2 ==$ " $*$ " **then**
        $subst\_AA$=stopS_1 +stopS_2
      else
        if $i - 3 > 0$ *AND* $j - 3 > 0$ **then**
          $subst\_AA = \sigma(AA_1, AA_2)$
        else
          $subst\_AA = +\infty$

$$C[i][j] = min \begin{cases} get\_C(i-3, j-3) + subst\_AA \\ get\_C(i-3, j) + stopS_1 + cost(" - ") \\ get\_C(i, j-3) + cost(" - ") + stopS_2 \\ \\ get\_C(i-3, j-2) + stopS_1 + cost("!") \\ get\_C(i-3, j-1) + stopS_1 + cost("!") \\ get\_C(i-2, j-3) + cost("!") + stopS_2 \\ get\_C(i-1, j-3) + cost("!") + stopS_2 \\ \\ get\_C(i, j-1) + cost(" - ") + cost("!") \\ get\_C(i, j-2) + cost(" - ") + cost("!") \\ get\_C(i-1, j) + cost("!") + cost(" - ") \\ get\_C(i-2, j) + cost("!") + cost(" - ") \\ \\ get\_C(i-1, j-1) + 2 * cost("!") \\ get\_C(i-1, j-2) + 2 * cost("!") \\ get\_C(i-2, j-1) + 2 * cost("!") \\ get\_C(i-2, j-2) + 2 * cost("!") \end{cases}$$

return C ;

L'algorithme 2 a la même complexité que l'algorithme 1 qui ne considère que le niveau NT (il nécessite simplement de considérer 15 cas au lieu de 3 pour chaque case de C). Ces deux algorithmes peuvent être étendus de manière à pénaliser les gaps selon une fonction affine. Il suffit pour cela d'utiliser trois tableaux C_I, C_D et C_S contenant les coûts des alignements des préfixes de S_1 et S_2 se terminant

respectivement par une insertion, une délétion ou une substitution (e.g. Kececioglu and Zhang, 1998).

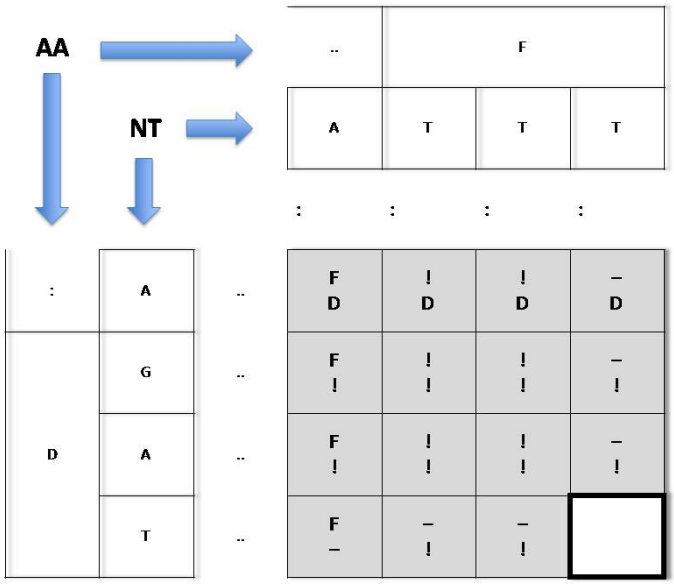

FIGURE 2.13 – **Relation entre les 15 mouvements possibles et l'alignement proposé.** *Supposons que l'algorithme produisant l'alignement ait conduit à la cellule en gras. Le prochain mouvement va se faire vers l'une des 15 cellules voisines colorées. Cette figure indique pour chacune de ces cellules, le site qui doit être ajouté à l'alignement en cours de construction si le mouvement menant à cette cellule est celui effectué.*

2.3.3.4 Alignement multiple de séquences codantes selon un modèle NT/AA intégrant les changements de cadre de lecture et les codons stops

L'alignement multiple A de n séquences S_1,...,S_n induit un alignement pour chaque paire de séquences S_i, S_j $(1 \leq i \leq j \leq n)$. Cet alignement s'obtient en supprimant toutes les autres séquences de A et tous les sites ayant une délétion à la fois pour S_i et S_j. Le coût d'un alignement multiple est souvent défini comme la somme des coûts des alignements qu'il induit pour chaque paire. Ce critère est appelé le "sum-of-pairs score" (ou "SP score"). Trouver l'alignement multiple optimal au sens du SP score est un problème *NP-complet* (Wang and Jiang, 1994), i.e. il n'existe pas d'algorithme polynomial garantissant de trouver la solution optimale.

Cependant, si l'on dispose de deux alignements A_1 et A_2 portant sur des ensembles disjoints de séquences S_1 et S_2, une variante de l'algorithme de "Needleman et Wunsch" permet d'obtenir un nouvel alignement A portant sur les séquences $S_1 \cup S_2$ et ayant le meilleur SP score, parmi les alignements qui induisent A_1 et A_2. Tout se passe comme si S_1 et S_2 étaient deux séquences et que leurs sites étaient des AA. Le coût de substitution d'un site en un autre est alors calculé de manière

à refléter la variation du critère SP, i.e. c'est la somme des coûts élémentaires pour transformer chaque AA du site de S_1 en chacun des AA du site de S_2. Un événement d'insertion (resp. délétion) revient à insérer un site ne contenant que des gaps dans S_2 (resp. S_1). Les coûts liés à l'extension de gaps d'un tel événement peuvent facilement être déduits à partir du nombre de gaps présents dans chaque site de S_1 et S_2. Par contre, il est difficile de déterminer le nombre de créations de gap induit par un tel évènement. Bien qu'il soit possible de calculer ce nombre de manière exacte (Kececioglu and Starrett, 2004) l'approximation, dite "pessimiste" de ce nombre, proposée par Altschul (1989) est beaucoup plus simple à mettre en œuvre et ne semble pas nuire à la qualité de l'alignement produit (Wheeler and Kececioglu, 2007). Il est important de noter que cette manière d'aligner A_1 et A_2 ne garantit pas l'obtention d'un alignement A qui soit un optimal global pour le SP score. Mais, elle constitue la base d'heuristiques efficaces permettant d'obtenir des alignements multiples de bonne qualité.

L'alignement multiple de séquences, produit par MACSE, repose sur une stratégie en deux étapes. Une première version de l'alignement est obtenue en intégrant progressivement les séquences de manière gloutonne (i.e. sans remettre en cause les choix précédents). Un arbre regroupant les séquences qui semblent les plus similaires est utilisé pour définir l'ordre dans lequel les séquences sont ajoutées. Puis, cet alignement est amélioré en cherchant une meilleure solution dans son voisinage ("hill-climbing"). Au cours de cette étape, l'alignement courant A est restreint de manière à produire deux alignement A_1 et A_2 portant chacun sur un sous-ensemble des séquences. Ces deux alignements sont ré-alignés, produisant un nouvel alignement A' qui devient l'alignement courant si son SP score est meilleur que celui de A. Cette stratégie est globalement similaire à celles utilisées dans ClustalW (Higgins et al., 1992), Muscle (Edgar, 2004b) et OPAL (Wheeler and Kececioglu, 2007). L'influence des choix possibles aux différentes étapes est analysée par Wheeler and Kececioglu (2007) et nous avons pris en compte leurs conclusions lors du développement de MACSE.

Les solutions que nous avons utilisées pour passer de l'alignement de deux séquences à l'alignement multiple ne sont pas réellement novatrices. Ce qui l'est, c'est d'avoir une solution d'alignement de deux séquences utilisant un modèle évolué NT/AA qui est compatible avec ces stratégies et qui permet d'obtenir un alignement multiple en un temps raisonnable. Je ne détaillerai donc pas davantage, dans cette section, ce passage à l'alignement multiple qui nous a pourtant pris beaucoup de temps (à Sébastien, puis à moi). En effet, les détails concernant la stratégie d'extension à l'alignement multiple ne sont pas complètement décrits dans les articles et les livres de références (loin s'en faut). De plus, certaines optimisations classiques ne sont plus possibles avec notre modèle, notamment du fait des changements de cadre de lecture, il nous a donc fallu trouver d'autres manières d'optimiser notre implémentation. Je pense cependant que ces efforts en valaient la peine. En effet, MACSE est déjà utilisé dans plusieurs projets internes et permet d'aligner automatiquement des séquences qui jusque-là ne pouvaient pas l'être. La section suivante fournit quelques exemples de résultats obtenus avec MACSE.

2.3.4 Validation et exemples d'applications biologiques

Cette section commence par comparer les performances de MACSE avec celles d'approches classiques sur des alignements de références ne contenant ni changement de cadre de lecture ni codon stop. Elle présente ensuite trois cas d'étude qui montrent l'intérêt de MACSE pour détecter des erreurs dans des séquences publiques, aligner des séquences de (pseudo)-gènes ou des séquences obtenues grâce aux nouvelles technologies de séquençages.

2.3.4.1 Validation sur des alignements de références

Il existe peu de jeux de données permettant de valider l'alignement de séquences NT codantes. Ceux construits par Carroll et al. (2007) à partir d'alignements d'AA issus des banques de données BAliBASE (Thompson et al., 2005), OXBench (Raghava et al., 2003), PREFAB (Edgar, 2004a) et SMART (Ponting et al., 1999) sont, à notre connaissance, les seuls de ce genre. La qualité d'un alignement produit par un logiciel est généralement évalué, au regard d'un alignement de référence, en considérant le pourcentage de paires de nucléotides placées dans un même site par les deux alignements (SP score). Nous avons calculé le SP score moyen des résultats obtenus i) en alignant les séquences NT avec MUSCLE (Edgar, 2004b) ; ii) en utilisant l'approche en trois étapes de TranslatorX (Abascal et al., 2010) lorsque la phase d'alignement des AA est effectuée par MUSCLE (Edgar, 2004b) ; et iii) en alignant les séquences avec MACSE. Ces résultats, obtenus à l'aide du programme baliscore, fournit par les auteurs de BALIBASE, sont résumés dans le tableau 2.1.

Le faible score de MUSCLE (SP moyen 69%), par rapport à ceux des deux autres approches (SP moyen 82% et 83%), confirme l'intérêt de prendre en compte la traduction en AA des séquences NT codantes que l'on souhaite aligner. Ces résultats montrent également que, lorsque les données ne contiennent ni changement de cadre de lecture ni codon stop, MACSE (SP moyen 82%) fournit des alignements de qualité comparable à ceux obtenus avec TranslatorX+MUSCLE (SP moyen 83%). Dans les jeux de données utilisés pour ces tests, le nombre de séquences à aligner peut aller jusqu'a près de 500 et l'alignement de référence peut contenir plus de 4 000 sites. Bien que plus lent que MUSCLE et TranslatorX+MUSCLE, MACSE est néanmoins capable de traiter de tels jeux de données en un temps raisonnable ne dépassant pas quelques heures (Tableau 2.2).

L'intérêt de MACSE peut sembler faible sur ce type de jeux de données ne contenant pas de changements de cadre de lecture. En pratique, on est rarement complètement sûr des séquences qu'on aligne. Il peut arriver que certaines, bien qu'issues de base de données fiables telle qu'EnsEMBL, contiennent des erreurs. La relative lenteur de MACSE est, à notre avis, largement compensée par sa capacité à produire des alignements de qualité en gérant correctement ce type d'erreurs.

TABLE 2.1 – **Evaluation de différentes stratégies d'alignements de séquences codantes suivant le SP score.**

méthode	1^{er} quartile	médiane	moyenne	$3^{ème}$ quartile
MUSCLE	56%	73%	69%	84%
TranslatorX + MUSCLE	76%	86%	83%	93%
MACSE	76%	86%	82%	94%

La qualité des alignements produits par trois méthodes différentes est évaluée sur 801 jeux de données distincts en fonction du pourcentage de paires de nucléotides situées dans une même colonne par cet alignement et par l'alignement de référence (SP score).

TABLE 2.2 – **Evaluation des temps de calculs de différentes stratégies d'alignements de séquences codantes**

Méthode	1^{er} quartile	médiane	moyenne	$3^{ème}$ quartile	max
MUSCLE	0s	2s	16s	11s	∼15min
TranslatorX + MUSCLE	2s	2s	4s	4s	122s
MACSE	45s	177s	∼24min	∼15min	12h

Temps de calculs des différentes méthodes, sur l'ensemble des 801 jeux de données testés.

2.3.4.2 Détection d'erreurs dans des banques de données publiques

Nous avons ré-aligné, à l'aide de MACSE, certains jeux de données exclus d'OrthoMaM (car ils n'avaient pas passé nos filtres de qualité). La Figure 2.14 montre les alignements obtenus, pour l'un de ces jeux de séquences, en utilisant translatorX+MUSCLE et en utilisant MACSE. MACSE détecte un changement de cadre de lecture chez deux primates (*Pan* et *Pongo*) et propose une traduction en AA de ces deux séquences en accord avec celles des autres primates. Grâce à l'introduction de ces changements de cadre de lecture, MACSE fournit un alignement beaucoup plus convaincant que celui proposé par translatorX+MUSCLE (la différence est flagrante au niveau AA). L'explication la plus probable est la présence d'une erreur dans ces deux séquences. Cette erreur ne conduisant pas, par hasard, à l'apparition de codon stop, elle passe inaperçue lors de l'alignement par translatorX+MUSCLE. En exploitant simultanément les niveaux NT/AA et en modélisant explicitement la possibilité de changement de cadre de lecture, MACSE offre la possibilité de détecter ce genre d'anomalies qui, présentes dans les séquences des banques publiques, peuvent passer inaperçues et fausser les analyses de nombreux chercheurs.

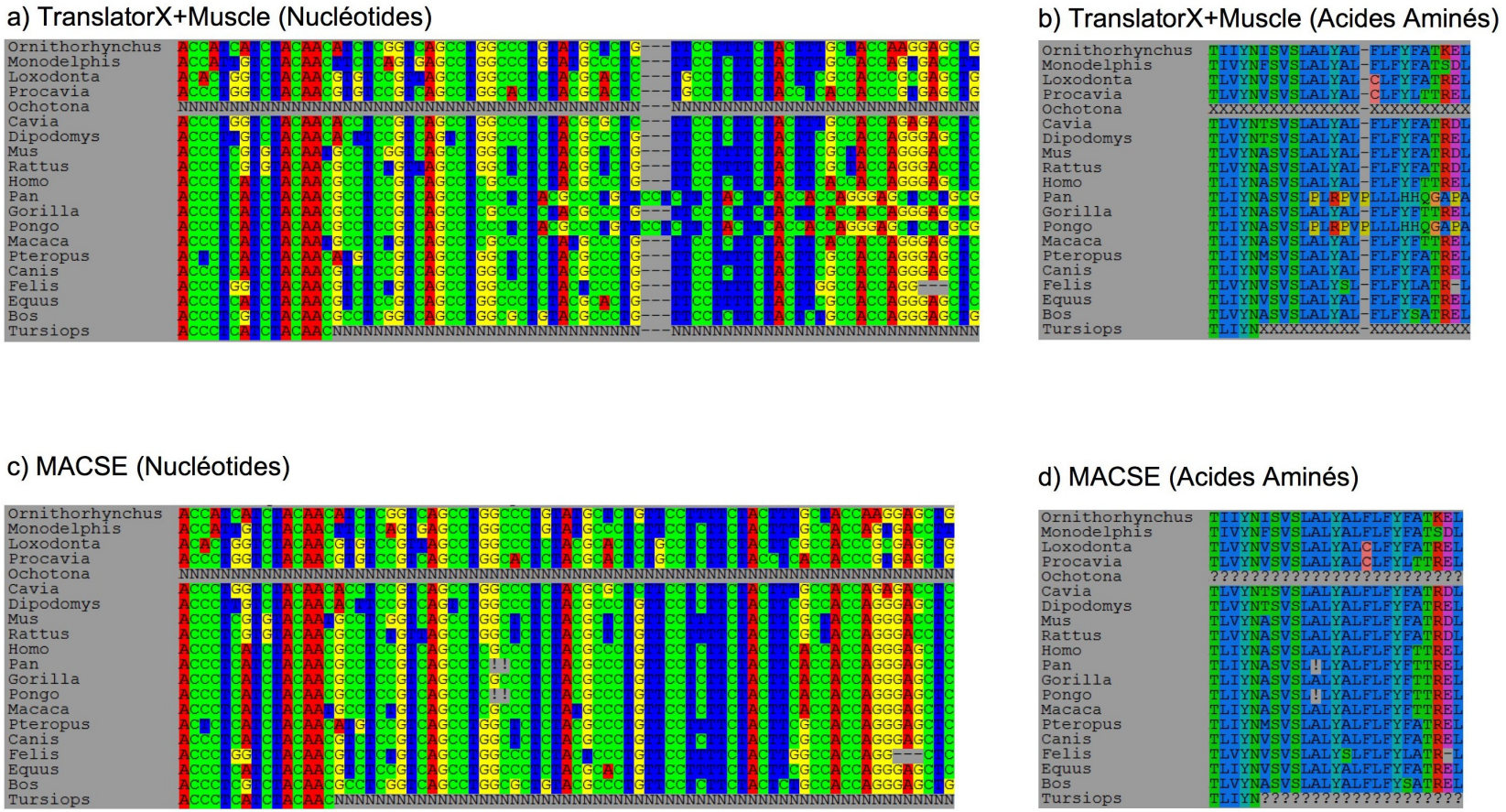

FIGURE 2.14 – **Mise en évidence d'une anomalie dans les séquences d'EnsEMBL à l'aide de MACSE.** *Cette figure présente l'alignement obtenu par translatorX+MUSCLE aux niveaux NT (a) et AA(b) et l'alignement produit par MACSE également aux niveaux NT(c) et AA(d). MACSE introduit un changement de cadre de lecture pour deux primates ce qui permet d'obtenir un alignement nettement plus satisfaisant où la similitude attendue entre séquences de primates est bien observée. L'explication la plus probable est que les séquences de ces deux primates contiennent une erreur (une Cytosine en trop). Les traductions en AA, qu'utilise translatorX, sont donc complètement faussées en aval de ces évènements et conduisent à un alignement erroné. Par contre, MACSE est capable de détecter et de gérer correctement ce problème.*

2.3.4.3 Alignement de séquences du (pseudo-)gène AMBN

Comme évoqué en introduction de cette section, le modèle utilisé par MACSE est particulièrement adapté pour aligner des pseudogénes avec les séquences correspondantes de gènes fonctionnels. A titre d'illustration nous avons utilisé MACSE, avec l'option "pseudo-gène", pour aligner des séquences du (pseudo-)gène AMBN qui n'est plus fonctionnel chez certaines espèces telles que les baleines. Cette option nécessite de fournir à MACSE deux fichiers en entrée, l'un contenant les séquences fonctionnelles, l'autre contenant les séquences de pseudo-gènes. MACSE utilise alors des pénalités différentes pour ces deux types de séquences afin de prendre en compte le fait que les codons stops et les changements de cadre de lecture sont plus fréquents dans les séquences non fonctionnelles. L'alignement produit par MACSE avec cette option contient plusieurs changements de cadre de lecture mis en évidence par l'introduction de "!" (Fig.2.15). Cet alignement peut notamment être utilisé pour des analyses sur la pression sélective s'exerçant sur ces séquences. Il suffit, par exemple, de supprimer les sites contenant un codon avec un "!" pour obtenir un alignement correct qui préserve la structure des codons et ne contient plus de caractères inhabituels.

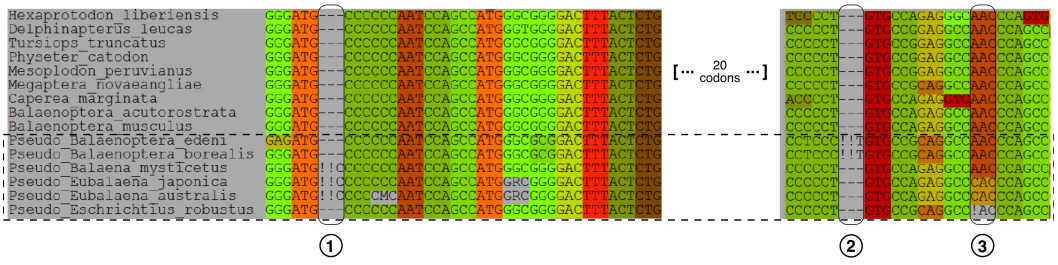

FIGURE 2.15 – **Alignement de séquences du (pseudo-)gène AMBN à l'aide de MACSE.** *Les "!" introduits par MACSE dans les séquences de Balaena et Eubalaena mettent en évidence l'ajout d'une Cytosine chez ces trois pseudo-gènes ①. Un évènement similaire est également détecté, à un autre site, chez les deux séquences de Balaenoptera : l'insertion d'une Thymidine semble ici l'explication la plus probable ②. Enfin, un "!" isolé est présent dans la séquence d'Eschrichtius indiquant une délétion d'un nucléotide chez ce pseudo-gène ③.*

2.3.4.4 Alignement de reads et détection d'erreurs de séquençage

Au vu de l'augmentation exponentielle du nombre de séquences produites par les technologies de séquençage à haut débit, il est évident que ces dernières vont jouer un rôle de plus en plus important dans de nombreuses études d'évolution moléculaire. Il devient donc essentiel de pouvoir aligner correctement ce type de séquences. Malgré la profondeur importante des séquençages obtenues avec ces nouvelles technologies, l'alignement de reads ou de contigs (cf section 1.2.3) issus des technologies 454 ou illumina, est compliqué par le fait qu'ils peuvent contenir des erreurs de séquençage (Kircher et al., 2009; Margulies et al., 2005). L'utilisation de MACSE peut aider à

détecter et gérer ces erreurs, que ce soit avant l'étape d'assemblage ou après.

Dans ce dernier exemple illustrant l'utilité de MACSE, nous l'avons utilisé pour aligner des reads issus du séquençage (454) du transcriptome de la gerboise (*Jaculus jaculus*) commandité par notre équipe. Nous avons assigné ces reads à des marqueurs d'OrthoMaM en utilisant une stratégie basée sur la similarité des séquences en utilisant le logiciel BLAST (Altschul et al., 1990). Il est alors possible d'utiliser MACSE pour aligner l'ensemble des séquences orthologues de rongeurs de chacun de nos marqueurs. On obtient ainsi des alignements confrontant les reads de la gerboise aux séquences orthologues, plus fiables, des cinq rongeurs modèles dont le génome est complètement assemblé et qui sont présents dans la version 59 d'EnSEMBL : la souris (*Mus Musculus*), le rat (*Rattus norvegicus*), le rat kangourou (*Dipodomys ordii*), le cochon d'Inde (*Cavia porcellus*), et l'écureuil (*Spermophilus tridecemlineatus*).

La Figure 2.16 montre une portion de l'alignement obtenu pour le gène "tmem214" dont l'identifiant EnsEMBL est, chez la souris, ENSG00000119777. Pour cet alignement MACSE introduit au total 4 changements de cadre de lecture qui se concentrent dans 3 des reads de la gerboise. Les deux premiers changements, détectés dans les séquences nommées read_5 et read_6, reflètent l'absence d'une Cytosine dans ces séquences. Le troisième changement est probablement dû à une erreur sur le nombre de Thymines. Idéalement MACSE devrait donc proposer "GAGC !TTTT" plutôt que "GAG !CTTTT". Mais notre modèle ne permet pas d'avoir ce niveau de précision puisque nos coûts ne sont basés que sur les traductions en AA, et que ces deux motifs conduisent à la même traduction "E !F". En cas de changement de cadre de lecture, MACSE ne fait donc que compléter le codon avec des " !" sans chercher quels sont les meilleures positions, parmi les trois possibles, pour ces " !". Il indique donc (son estimation de) la localisation des erreurs à deux bases près. Enfin, le dernier cas correspond à une Guanine absente du read_7. L'alignement proposé par MACSE est tout à fait convaincant malgré la présence de ces multiples erreurs. On peut donc penser que l'utilisation de MACSE, en permettant de détecter et de gérer ces erreurs, pourrait améliorer la phase d'assemblage faite par des outils tels que CAP3 (Huang and Madan, 1999) ou miraEST (Chevreux et al., 2004). Une telle utilisation est particulièrement pertinente pour des zones de faible couverture où MACSE pourrait permettre de choisir entre plusieurs reads proposant des séquences alternatives. On peut également l'utiliser pour identifier les reads contenant le plus d'erreurs (e.g. le read_05 dans la Figure 2.16) et retirer ces reads avant d'effectuer la phase d'assemblage. Enfin, les alignements de qualité qu'est capable de produire MACSE, même en présence d'erreurs de séquençage, sont particulièrement utiles dans les analyses phylogénomiques basées sur des alignements d'AA qui permettent d'inférer des relations évolutives anciennes (Delsuc et al., 2008).

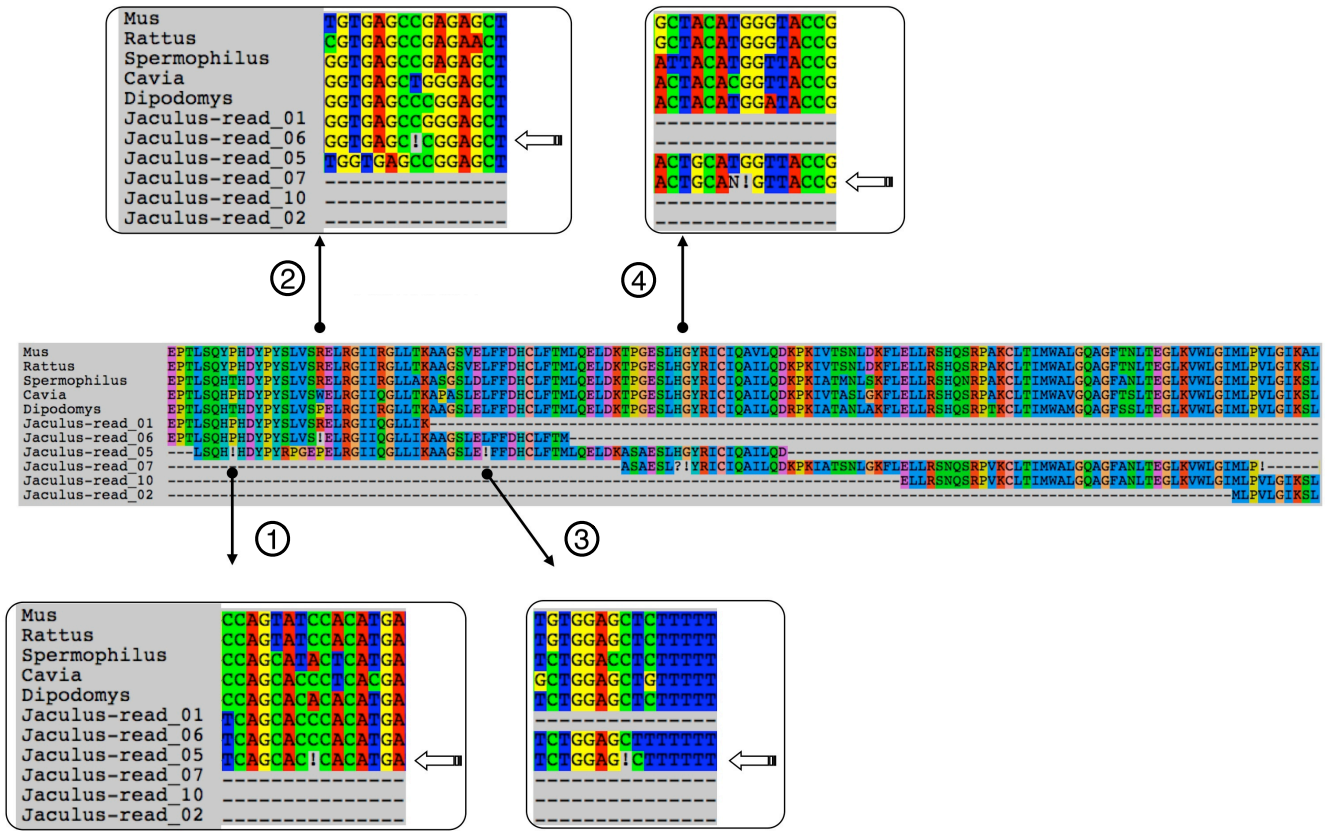

FIGURE 2.16 – **Alignement de reads issus du séquençage 454 d'un transcriptome de rongeur.** *Dans cet exemple les séquences orthologues du gène tmem214 de cinq rongeurs modèles sont alignées avec les séquences produites par le séquençage 454 du transcriptome d'un autre rongeur : la gerboise (Jaculu jaculus). L'alignement protéique produit par MACSE, au centre de la figure, détecte quatre changements de cadre de lecture (indiqué par des "!"). Pour chacun d'eux, une vision locale de l'alignement nucléotidique est fournie. Les "!" de ces alignements nucléotidiques indiquent les erreurs de séquençage qui se sont probablement produites.*

2.4 Conclusion et perspectives

L'obtention d'alignements de séquences orthologues est l'une des premières étapes de nombreuses analyses moléculaires et phylogénétiques. La qualité de cette étape est primordiale et conditionne, en partie, la validité des résultats obtenus ultérieurement sur la base de ces données. Cela explique, l'effort important que nous y avons porté à la fois en développant et en maintenant OrthoMaM et en améliorant les solutions algorithmiques et logicielles permettant d'aligner des séquences codantes. Ces deux projets sont d'ailleurs liés puisque nous avons l'intention d'utiliser MACSE pour produire les alignements des prochaines versions d'OrthoMaM.

L'extension de mes travaux de recherche concernant l'alignement de séquences portent sur deux aspects. Le premier est lié à l'ajustement automatique des paramètres. En effet, les résultats des alignements sont sensibles aux paramètres utilisés (notamment les coûts d'ouverture et d'extension des gaps). En ajoutant de nouveaux paramètres sur lesquels nous avons encore peu de recul, MACSE complique encore les choses. Nos paramètres par défaut semblent satisfaisants, mais les travaux de (Wheeler and Kececioglu, 2007) montrent que la détermination des paramètres optimaux permettraient d'améliorer énormément les performances des logiciels d'alignement. Le second aspect concerne l'optimisation de la mémoire, qui est souvent le facteur limitant. Bien qu'il existe des solutions permettant d'aligner deux séquences avec un espace mémoire linéaire (Chao et al., 1994), elles ne sont pas utilisées pour l'alignement multiple car elles augmentent significativement les temps de calcul. Cependant, nous pensons qu'il est possible de trouver un compromis permettant de réduire sensiblement l'espace mémoire, sans trop augmenter les temps de calcul. Les travaux de Newberg (2008) offrent, sur ce sujet, des pistes intéressantes que nous pensons creuser.

Plusieurs développements d'OrthoMaM sont en cours. Ils concernent notamment la mise en place de "Web services" et l'optimisation des procédures de mises à jour (afin de pouvoir gérer la forte augmentation du nombre d'espèces qui ne serait tarder). De plus, la thèse de Jonathan sur la conversion génique biaisée ne fait que débuter (il termine juste sa première année). Un des objectifs de sa thèse est de proposer, et d'implémenter, un modèle d'évolution des séquences qui prenne en compte les enrichissements en GC dus à la conversion génique biaisée. L'utilisation d'un tel modèle sur les données d'OrthoMaM devrait permettre de mieux comprendre l'impact et l'importance de la conversion génique biaisée sur l'évolution des séquences. La difficulté réside principalement dans le fait que l'enrichissement est localisé sur une portion de la séquence et affecte simultanément plusieurs sites. Ce modèle ne peut donc pas faire l'hypothèse d'une évolution indépendante des sites. Or cette hypothèse d'indépendance est fondamentale dans les modèles actuels, c'est elle qui permet de modéliser efficacement l'évolution des séquences et d'identifier en un temps raisonnable les scénarios évolutifs parmi les plus vraisemblables. Nous pensons néanmoins pouvoir contourner ce problème en utilisant une classe d'algorithmes connue sous le terme générique d'ABC (pour Approximate Bayesian Computation) qui peut s'appliquer aux cas où il n'existe pas d'expression suffisamment explicite de la fonc-

tion de vraisemblance. Ces méthodes, très utilisées en génomique des populations, sont encore peu appliquées en phylogénie.

Les progrès en matière de séquençage ont complètement changé la manière dont nous obtenons et gérons des alignements de séquences orthologues. En effet, l'alignement qui était souvent vérifié par un expert n'est, de plus en plus, qu'une étape dans un "pipe-line" plus large. Cette évolution est très nette au sein de notre équipe. Il y a trois ans, il était fréquent de consacrer plusieurs semaines pour obtenir 3 ou 4 séquences d'espèces d'intérêts. Ces séquences étaient vérifiées individuellement et leur alignement, avec d'autres séquences publiques, était corrigé manuellement par un expert. Aujourd'hui, nos sous-traitants peuvent nous fournir le séquençage du transcriptome complet de ces espèces en quelques jours. Les milliers d'alignements obtenus ne peuvent plus être vérifiés manuellement. Heureusement, l'utilisation de MACSE lors de la phase d'alignement devrait nous permettre de détecter, de manière automatique, une part importante des erreurs contenues dans ces séquences. Il est clair que ces nouvelles technologies de séquençage vont également nous amener à changer la manière dont nous allons construire et utiliser OrthoMaM. Nous ne pourrons pas gérer des milliers de génomes comme nous en gérons une trentaine. Mais ce changement d'échelle, même s'il pose des problèmes méthodologiques, est surtout une formidable opportunité de mieux comprendre le monde vivant qui nous entoure.

Chapitre 3

Méthodes de super-arbre

If all mankind minus one were of one opinion, and only one person were of the contrary opinion, mankind would be no more justified in silencing that one person, than he, if he had the power, would be justified in silencing mankind.

John Stuart Mill

Sommaire

Les méthodes de super-arbre permettent de combiner, en une seule phylogénie, des histoires évolutives distinctes portant sur différents groupes de taxons. Elles peuvent être utilisées aussi bien pour combiner le signal de nombreux marqueurs phylogénétiques que pour proposer une synthèse de phylogénies publiées dans la littérature. Ce chapitre présente une partie de mes travaux sur les super-arbres. Dans un premier temps, nous y décrivons deux propriétés théoriques, que des méthodes de super-arbre visant à inférer l'Arbre de la Vie devraient satisfaire (Ranwez et al., 2007a). Puis, nous présentons la méthode *PhySIC_IST*, qui est basée sur ces propriétés (Scornavacca et al., 2008) et résumons des résultats concernant la phylogénie

des Triticea (Escobar et al., soumis) obtenus à l'aide de cette méthode. Enfin nous présentons les principales fonctionnalités de PhyloExplorer (Ranwez et al., 2009), un serveur Web permettant de gérer efficacement une collection d'arbres, très utile dans le cadre d'analyse de type super-arbre.

3.1 Brève introduction aux méthodes de super-arbre

L'essentiel de mes travaux sur les méthodes de super-arbre s'inscrit dans une collaboration avec Vincent Berry qui est professeur au LIRMM (le Laboratoire d'Informatique et de Micro-électronique de Montpellier). Ces travaux ont fait l'objet d'un soutien financier de la part de l'Université de Montpellier (appel d'offe interne) puis de la part de l'Agence Nationale pour la Recherche (ANR). Ce projet ANR, intitulé Phyl-Ariane (`http://www.lirmm.fr/phylariane/index.php`), a pour principal objectif le développement d'outils méthodologiques permettant d'inférer de manière fiable une partie importante de l'Arbre de la Vie. Outre l'ISEM et le LIRMM, ce projet implique également le Laboratoire Biométrie et Biologie Évolutive (LBBE) de Lyon.

L'université de Montpellier a également soutenu ce projet au travers de l'attribution d'une "bourse président". Cette bourse a permis de financer la thèse de Céline Scornavacca dans le cadre d'un co-encadrement LIRMM-ISEM où Vincent Berry et moi-même étions les deux encadrants de proximité et Olivier Gascuel le directeur de thèse. Une part importante des travaux décris dans ce chapitre sont liés à cette thèse (Scornavacca, 2009) qui contient un excellent état de l'art sur les méthodes de super-arbre, qu'il me semble inutile de reprendre ici. Afin de comprendre le positionnement de nos travaux, cette section présente néanmoins les grandes familles de méthodes de super-arbre existantes en insistant sur la méthode de référence à laquelle nous nous comparons (MRP : *Matrix Representation with Parcimony*).

3.1.1 Trois approches pour gérer les conflits topologiques

Depuis la publication originale du super-arbre des Primates (Purvis, 1995a), les super-arbres sont devenus de plus en plus populaires, à tel point qu'un livre entier leur a été consacré (Bininda-Emonds, 2004). L'une des difficultés inhérentes aux méthodes de super-arbre est l'utilisation d'arbres sources incongruents, c'est-à-dire en désaccord sur la position phylogénétique de certains taxons. Selon leur manière de gérer ces topologies conflictuelles, les méthodes de super-arbre se divisent en trois grands types. Dans le premier, les topologies sources incongruentes ne sont pas assemblées. Dans les deux autres types, les topologies sources sont toujours assemblées – quel que soit leur degré de congruence –, mais selon des philosophies distinctes ; les incongruences sont traitées par une procédure de "vote" dans le deuxième type, tandis qu'elles sont gérées par une procédure de "véto" dans le troisième.

Les approches pionnières de super-arbres telles que la méthode Build (Aho et al., 1981) et le consensus strict (Gordon, 1986) sont classées dans la première famille. Bien que constituant une étape importante de l'histoire des super-arbres,

Bininda-Emonds les désigne comme étant « of limited utility. As most systematists know, phylogenies usually conflict with one another » (Bininda-Emonds, 2004, p. 4). D'éventuelles incongruences peuvent effectivement émerger lorsque le signal phylogénétique diffère d'un jeu de données initial à l'autre. Par exemple, lorsque les arbres sources sont des arbres de gènes, le signal phylogénétique principal peut être faible ou brouillé par des problèmes méthodologiques, des hybridations entre espèces (surtout chez les plantes), ou des évènements macro évolutifs de transferts, duplications et pertes de gènes (cf section1.3).

Dans la deuxième famille de méthodes, dite *de vote*, une décision est prise en faveur de l'une ou l'autre des résolutions alternatives observées parmi les arbres sources. Ce choix se fait sur la base d'un critère d'optimisation qui varie d'une méthode à l'autre. Ces méthodes sont supposées "résoudre" les conflits (Thorley and Wilkinson, 2003). Une telle stratégie est adaptée aux études dont l'objectif est d'extraire le signal phylogénétique congruent d'un lot d'arbres sources contenant des informations de fiabilité variable. Pour cela, il est raisonnable de "faire voter" les topologies sources, et ainsi d'élire les scénarios évolutifs les plus soutenus. Dans ce contexte, l'approche la plus répandue est celle dite de "Représentation Matricielle avec Parcimonie (ou MRP)" (Baum, 1992; Ragan, 1992; Baum and Ragan, 2004). Malgré de nombreuses critiques concernant cette méthode, elle reste la méthode de référence en matière de super-arbre. En effet, elle est de loin la plus employée et, malgré ses défauts, fournit en pratique d'assez bons résultats. Nous comparerons donc les résultats de notre méthode (*PhySIC_IST*) à ceux de MRP, ce qui explique que la section 3.1.2 détaille le principe et les limites de cette méthode.

Dans la troisième famille de méthodes de super-arbre, dite *de véto*, le message phylogénétique de chaque arbre source est respecté. Ainsi, un groupement de taxons (un *clade*) est retenu dans le super-arbre si, et seulement si, les topologies sources sont unanimement en accord avec sa présence. Le super-arbre ne peut donc pas contenir des clades auxquels un des arbres sources pourrait s'opposer. Pour cette raison, de telles méthodes proposent des super-arbres contenant des *irrésolutions* (Bryant, 2002) – i.e. des nœuds internes ayant plus de deux fils – ou excluant certains taxons problématiques (Berry and Nicolas, 2004). Dans la terminologie de Thorley and Wilkinson (2003), ces méthodes "retirent" les conflits. La principale application des méthodes de super-arbre de type *véto* est de construire l'Arbre de la Vie. Atteindre cet objectif requiert de partir de phylogénies sources bien établies, quoique fragmentaires, pour obtenir des super-arbres toujours plus grands, à la fois taxonomiquement représentatifs et fortement soutenus. Tous les arbres sources étant supposés fiables, le super-arbre reconstruit ne doit pas favoriser une topologie initiale au détriment d'une autre. Au contraire, il doit indiquer les zones de l'Arbre de la Vie où des incertitudes subsistent. Plusieurs méthodes de super-arbre suivant cette approche par véto ont été récemment proposées. Toutes s'inspirent des *méthodes de consensus*, i.e. des méthodes qui opèrent sur des arbres sources ayant le même ensemble de feuilles. Il existe notamment, des extensions du consensus strict (Gordon, 1986; Huson et al., 1999), du consensus semi-strict (Goloboff and Pol, 2002), et du sous-arbre d'accord maximum (Berry and Nicolas, 2004).

3.1.2 La méthode MRP : Matrix Representation with Parcimony

Cette méthode procède en deux étapes. La première transforme la collection d'arbres sources en une matrice de caractères binaires où chaque ligne représente un taxon et chaque colonne une information topologique élémentaire d'un arbre source. La seconde étape consiste à rechercher *l'arbre le plus parcimonieux* pour cette matrice. L'arbre le plus parcimonieux représente le scénario évolutif qui permet, avec un minimum de mutations, d'expliquer comment une séquence ancestrale a pu évoluer de manière à engendrer les séquences observées chez les différents taxons.

On peut utiliser MRP avec différentes représentations matricielles suivant les informations topologiques élémentaires que l'on considère. La représentation la plus courante s'appuie sur les "clades" (resp. les "bipartitions") pour les arbres enracinés (resp. non racinés). Dans le cas d'arbres enracinés, chaque nœud définit un sous-ensemble de taxons (correspondant à ces descendants) que l'on appelle *clade*. Pour les arbres non enracinés, chaque branche de l'arbre définit une partition des taxons en deux groupes (selon qu'ils sont d'un côté ou de l'autre de la branche) que l'on appelle *bipartition*. Steel et al. (2000) ont énoncé une liste de propriétés simples que toute méthode d'inférence de super-arbres devrait vérifier. Cependant, dans le même article, ils ont prouvé qu'aucune méthode basée sur des arbres sources non enracinés ne pouvait satisfaire simultanément toutes ces propriétés. De manière surprenante, cette restriction ne s'applique pas pour des méthodes utilisant des arbres enracinés (e.g., la méthode Mincut de Semple and Steel, 2000). Cette remarque explique d'ailleurs le fait que nos travaux se focalisent sur l'utilisation d'arbres enracinés.

La Figure 3.1 montre un exemple d'une collection d'arbres sources (ou *forêt*), la matrice binaire permettant de coder ses clades et le super-arbre proposé par MRP pour cette forêt. Pour cet encodage, la matrice contient une ligne par taxon et une colonne par clade. Notez que cette matrice ne contient pas les *clades triviaux*, i.e. ceux correspondants aux feuilles (qui ne contiennent qu'un seul taxon) et aux racines des arbres sources (qui contiennent tous les taxons de l'arbre). En effet, ces clades n'ont pas d'impact sur le score de parcimonie et sont donc inutiles. Pour les autres clades, la valeur d'une cellule de la matrice binaire est : 1 si le taxon est dans le clade ; 0 si le taxon est ailleurs dans l'arbre contenant ce clade et ? si le taxon n'est pas dans cet arbre. La recherche de l'arbre le plus parcimonieux à partir de cette matrice ne se fait que sur l'espace des arbres complètement résolus (ou *arbres binaires*), i.e. ceux dont tous les nœuds internes ont deux fils. Quand plusieurs arbres de ce type ont la même valeur minimale de parcimonie, l'arbre renvoyé par MRP est leur *strict consensus* i.e. l'arbre ne contenant que les clades présents dans chacun de ces arbres optimaux. Certains auteurs ont proposé des solutions alternatives (e.g. Wilkinson et al., 2007) mais l'utilisation du consensus strict reste de loin la solution la plus utilisée.

La recherche de l'arbre le plus parcimonieux est relativement justifiée lorsque la matrice binaire représente des caractères morphologiques ou des évènements génomiques rares pour lesquels il peut sembler plus réaliste qu'ils soient apparus une seule fois chez l'ancêtre commun de deux taxons plutôt que deux fois indépendamment

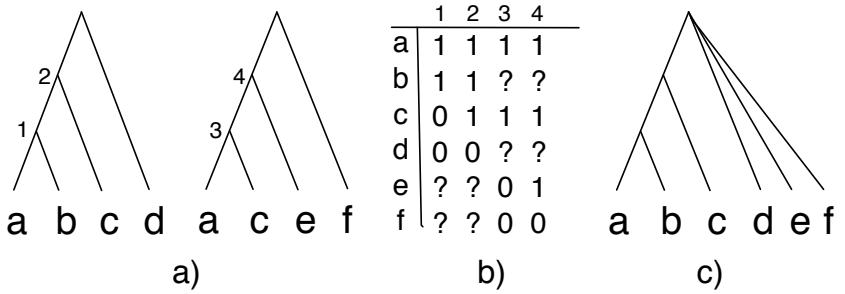

FIGURE 3.1 – **Exemple d'application de la méthode MRP.** *Les clades de chacun des nœuds internes des deux arbres sources a) sont encodés de manière binaires dans la matrice b). Le strict consensus c) des arbres les plus parcimonieux de cette matrice correspond au super-arbre proposé par MRP.*

chez chacun de ces taxons. Dans le cas où le caractère représente une information topologique d'un arbre source la justification du critère de parcimonie est nettement moins claire. Les travaux de Bruen and Bryant (2008) ont cependant permis de mettre en évidence un lien entre le critère de parcimonie et une distance entre arbres. Un tel lien était déjà connu dans le cas d'un codage binaire particulier basé sur les "triplets" d'un arbres, i.e. l'ensemble des résolutions qu'il induit sur les sous-groupe de trois taxons. Avec cet encodage, l'arbre le plus parcimonieux correspond à l'arbre médian (au sens de la distance de triplets) de la forêt d'arbres sources (Wilkinson et al., 2001, 2005). Cependant, dans le cas de triplets cette manière de chercher l'arbre médian est inefficace car la matrice binaire utilisée est creuse (elle contient essentiellement des " ?"). Nous avons d'ailleurs récemment proposé une méthode plus efficace pour traiter ce problème (Ranwez et al., 2010).

La vision globale des topologies sources permet à MRP de proposer des groupements d'espèces qui ne sont présents dans aucun des arbres initiaux (pris individuellement) mais qui sont collectivement induits par plusieurs de ces arbres. Cette capacité à faire émerger des groupements "induits" fait tout l'intérêt des méthodes de super-arbre. Malheureusement, MRP peut aussi proposer des groupements qui contredisent l'ensemble des arbres sources (Goloboff and Pol, 2002; Goloboff, 2005). Ce défaut majeur survient même dans le cas plus simple où tous les arbres sources ont le même ensemble de feuilles. D'autres critiques concernent la tendance de MRP à favoriser certains arbres sources et certains clades (Purvis, 1995b). Tout comme Pisani and Wilkinson (2002) nous restons convaincus que « MRP may suffer from potentially serious but poorly understood biases and from its potential to produce unjustified new groups. We consider that the properties of MRP [...] should be better understood before MRP can be reasonably adopted as a method of choice for supertree construction ».

3.2 Propriétés désirables pour des méthodes de super-arbre

3.2.1 Motivations

Une utilisation phare des méthodes de super-arbre est d'assembler des phylogénies fiables, telles que celles publiées récemment dans différentes revues scientifiques, afin de pouvoir fournir de manière automatique un état de l'art sur nos connaissances actuelles de l'Arbre de la Vie. Il est clair que dans ce contexte, il n'est pas souhaitable d'utiliser une méthode telle que MRP, qui risque d'introduire dans le résultat final des informations qui sont absentes des données sources, ou pire, des informations qui sont contredites par l'ensemble de ces données sources. Au contraire, on souhaite, que le super-arbre produit ait la propriété de ne pas contredire les clades sources (*propriété de non-contradiction*, notée PC) et que tous ses clades soient présents ou induits par ceux des arbres sources (*propiété d'induction*, notée PI). Cette section introduit le vocabulaire et les notations nécessaires pour définir de manière formelle PC et PI. Des exemples simples illustrent aussi la pertinence de ces deux propriétés et permettent de les comparer avec des propriétés proposées dans un contexte similaire.

En effet, avant d'entamer le développement d'une nouvelle méthode de super-arbre, il nous a semblé important de préciser les propriétés que celle-ci devait respecter. La formalisation et l'étude de ces propriétés font donc partie des premiers travaux de recherche que j'ai menés au sein de l'ISEM. Ceci s'est fait, essentiellement, en collaboration avec Vincent Berry (LIRMM) et Emmanuel Douzery (responsable de l'équipe PhylMol à l'ISEM). Céline Scornavaca a naturellement rejoint ce petit groupe de travail dès le début de sa thèse. Elle fait donc évidemment partie des co-auteurs de l'article publié dans *Systematic Biology* qui présente ces travaux (Ranwez et al., 2007a).

3.2.2 Définitions et notations

Pour un arbre enraciné ayant trois feuilles a, b, c, il n'existe que trois topologies binaires possibles appelées *triplets* et notées $ab|c$, resp. $ac|b$, resp. $bc|a$ en fonction du seul clade non trivial de cet arbre ($\{a, b\}$, resp. $\{a, c\}$, resp. $\{b, c\}$). Il existe également une topologie irrésolue : l'arbre étoile, i.e. l'arbre topologiquement non informatif constitué d'un seul noeud interne directement relié aux trois feuilles. Etant donné un triplet t, on note $\bar{t}$ n'importe lequel des deux autres triplets ayant les mêmes feuilles. Un arbre T de plus de trois feuilles peut être représenté par l'ensemble des triplets homéomorphiques aux sous-arbres de T contenant trois feuilles (Semple and Steel, 2003). Cette représentation est largement utilisée dans le contexte des super-arbres. Dans la suite, l'ensemble de triplets équivalent à T sera noté $tr(T)$. La Figure 3.2 fournit un exemple de l'ensemble $tr(T)$ obtenu pour un arbre T à 5 feuilles. Cette notation se généralise naturellement à une forêt $\mathcal{F}$: $tr(\mathcal{F}) = \bigcup_{T_i \in \mathcal{F}} tr(T_i)$. Il est possible que $tr(\mathcal{F})$ contienne à la fois t et $\bar{t}$, il suffit pour cela que des arbres sources

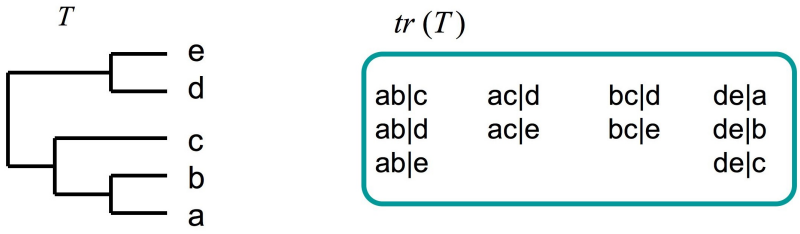

FIGURE 3.2 – **Décomposition d'un arbre en triplets.**

contiennent des résolutions contradictoires d'un même triplet.

Définition 3.2.1 – **Représentation et compatibilité** : *Etant donné $\mathcal{R}$ un ensemble de triplets, un arbre T représente $\mathcal{R}$ ssi $\mathcal{R} \subseteq tr(T)$. Un ensemble de triplets $\mathcal{R}$ est compatible ssi il existe au moins un arbre qui le représente.*

Définition 3.2.2 – **Induction, cas compatible** : *Soit $\mathcal{R}$ un ensemble compatible de triplets. On dit que $\mathcal{R}$ induit le triplet t ($\mathcal{R} \vdash t$) ssi $\mathcal{R} \cup \bar{t}$ est incompatible. Une définition alternative de $\mathcal{R} \vdash t$ consiste à dire que t doit être présent (*i.e.*, $t \subseteq tr(T)$) dans chaque arbre T qui représente $\mathcal{R}$ (Grunewald et al., 2007).*

Par exemple, si $\mathcal{R} = \{ab|c, bc|d\}$ alors tout arbre T qui représente $\mathcal{R}$ contient également le triplet $ac|d$ (i.e. $ac|d \in rt(T)$). Ce qui peut s'écrire : $\{ab|c, bc|d\} \vdash ac|d$ Pour s'en convaincre il suffit de dessiner l'arbre correspondant au triplet $bc|d$, puis d'essayer d'ajouter le taxon a sur cet arbre. La seule solution, pour que l'arbre final contienne également $ab|c$, est de relier a à la branche qui part de b ; l'arbre obtenu contient alors le triplet $ac|d$. Les premiers à avoir étudié ces propriétés d'induction sont Bandelt and Dress (1986) et Dekker (1986). L'exemple précédent est un cas où l'induction se fait à partir de deux triplets. Mais il existe des inductions complexes qui impliquent un nombre quelconque de triplets et qui ne peuvent pas être déduites de règles plus simples (Bryant and Steel, 1995). Ces triplets induits sont capitaux dans l'inférence de super-arbres, ils permettent de déduire des nouvelles informations en combinant celles présentes dans plusieurs arbres sources.

Définition 3.2.3 – **Fermeture** : *Soit $\mathcal{R}$ un ensemble de triplets, la fermeture (closure) de $\mathcal{R}$, notée $cl(\mathcal{R})$, est définie de la manière suivante :* $\mathbf{cl}(\mathcal{R}) = \{ab|c \text{ tel que } \mathcal{R} \vdash ab|c\}$.

Pour pouvoir définir PC et PI, nous avons généralisé la définition 3.2.2, proposée par Grunewald et al. (2007), au cas où l'ensemble de triplets n'est pas forcément compatible.

Définition 3.2.4 – **Induction, cas général** : *Soit $\mathcal{R}$ un ensemble de triplets et t un triplet. On dit que $\mathcal{R}$ induit t ($\mathcal{R} \vdash t$) ssi $\exists \mathcal{R}' \subseteq \mathcal{R}$ tel que $\mathcal{R}'$ est compatible et $\mathcal{R}' \vdash t$. L'ensemble des triplets induits par $\mathcal{R}$ sera noté $ind(\mathcal{R})$. Notons que lorsque $\mathcal{R}$ est compatible, $cl(\mathcal{R}) = ind(\mathcal{R})$.*

Définition 3.2.5 – **Identification d'un arbre** : *Soit $\mathcal{R}$ un ensemble compatible de triplets. On dit que $\mathcal{R}$ identifie un arbre T ssi $cl(\mathcal{R}) = tr(T)$.*

Il est clair qu'un ensemble incompatible n'identifie donc aucun arbre. Notons que si $\mathcal{R}$ est un ensemble compatible de triplets alors il y a au moins un arbre qui représente $\mathcal{R}$, mais cela n'est pas suffisant pour assurer que $\mathcal{R}$ *identifie* un arbre particulier.

3.2.3 Propriétés de non-contradiction (PC) et d'induction (PI)

Dans le cas où $\mathcal{F}$ identifie un arbre T, T est un super-arbre idéal pour représenter $\mathcal{F}$. En pratique il est peu fréquent qu'une collection d'arbres sources $\mathcal{F}$ soit compatible (Bininda-Emonds, 2004, p. 4) ; or, même dans de tels cas, elle n'identifie pas nécessairement un arbre particulier. Dans le cas général où $tr(\mathcal{F})$ n'est pas forcément compatible, il est cependant possible qu'un sous-ensemble $\mathcal{R}_T$ de $tr(\mathcal{F})$ identifie un arbre T. Dans ce cas, chaque information topologique de T est présente, directement ou de manière induite, dans $\mathcal{F}$. Ceci fait de T un bon candidat pour représenter $\mathcal{F}$. Néanmoins, pour que T soit un super-arbre valable, dans le contexte de projets du type Arbre de la Vie, il est également souhaitable que ses triplets ne contredisent pas d'autres triplets de $tr(\mathcal{F})$ sur les mêmes ensembles de feuilles. Si tel est le cas, T représente alors un sous-ensemble consensuel d'informations topologiques présentes dans les arbres sources, tandis que les informations conflictuelles de $\mathcal{F}$ ne sont pas présentes dans T (soit du fait d'irrésolutions dans T soit du fait de l'absence de certains taxons).

Définition 3.2.6 – $\mathcal{R}_\mathbf{T}$: *Soit T un arbre, et $\mathcal{F}$ un ensemble d'arbres. On définit $\mathcal{R}_\mathbf{T}(\mathcal{F}) = \{ab|c \in tr(\mathcal{F})$ tel que $\{ab|c, ac|b, bc|a\} \cap tr(T) \neq \emptyset\}$. En l'absence d'ambiguïté sur $\mathcal{F}$ on notera simplement $\mathcal{R}_T$.*

Notons qu'il est possible que $\mathcal{R}_T(\mathcal{F})$ soit incompatible. C'est notamment le cas dès que T contient un triplet qui est résolu différemment dans deux arbres de $\mathcal{F}$. A l'aide de ces notations, il est maintenant possible de définir de manière formelle les propriétés évoquées ci-dessus.

Définition 3.2.7 – **PI et PC** : *Soit $\mathcal{F}$ un ensemble d'arbres et T un super-arbre, on dit que :*

- *T vérifie* **PI** *pour $\mathcal{F}$ ssi $\forall t \in tr(T)$, $\mathcal{R}_T \vdash t$.*
- *T vérifie* **PC** *pour $\mathcal{F}$ ssi $\forall t \in tr(T)$, $\mathcal{R}_T \nvdash \bar{t}$.*

PI et PC sont des propriétés pertinentes dans le sens où, dès qu'un arbre T vérifie à la fois PI et PC, cela garantit que $\mathcal{R}_T$ représente une partie de $tr(\mathcal{F})$ qui correspond exactement à un arbre.

Proposition 3.2.8 *Soit $\mathcal{F}$ un ensemble d'arbres et T un super-arbre, $\mathcal{R}_T$ identifie T (ie $cl(\mathcal{R}_\mathcal{F}) = tr(T)$) ssi T verifie PI et PC pour $\mathcal{F}$.*

La preuve de cette proposition, ainsi que celles des autres résultats théoriques de cette section, se trouve dans l'article Ranwez et al. (2007a).

Définition 3.2.9 – **Contradiction directe** : *Un arbre T* contredit directement *un ensemble de triplets $\mathcal{R}$ ssi $\exists t \in tr(T)$ tel que $\bar{t} \in \mathcal{R}$. En particulier, pour un ensemble d'arbres sources $\mathcal{F}$, si T contredit directement $\mathcal{R} = tr(\mathcal{F})$, alors on dit que T contredit $\mathcal{F}$.*

Lemme 3.2.10 *Si T est un arbre qui ne contredit pas directement $\mathcal{F}$, alors T vérifie les trois propriétés suivantes :*
1. $\mathcal{R}_T \subseteq tr(T)$; *2. $\mathcal{R}_T$ est compatible ;* *3. PC.*

Ce lemme est fondamental, car il garantit que l'absence de contradiction directe entre $tr(T)$ et $rt(\mathcal{F})$ assure l'absence totale de contradiction, même indirecte, entre ces deux ensembles. Or il est très simple de vérifier l'absence de contradiction directe (il suffit de s'assurer qu'il n'existe aucun triplet t tel que $t \in tr(T)$ et $\bar{t} \in tr(\mathcal{F})$) alors que l'absence de contradiction indirecte est habituellement beaucoup plus difficile à vérifier (à cause des règles d'induction).

3.2.4 Liens avec d'autres propriétés souhaitables des super-arbres

Des propriétés similaires à PI et PC sont décrites dans (Goloboff and Pol, 2002, p.519) : « the property of [the supertree] displaying $ab|c$ if it is found in some input tree or implied by some combination of input trees and no input tree or combination of input trees displays or implies $ac|b$ or $bc|a$. » Ces propriétés, dont la pertinence est également soulignée par Grunewald et al. (2007), peuvent être décrites dans notre formalisme comme suit :

$$- \ \mathbf{PI'} : \forall t \in T, tr(\mathcal{F}) \vdash t \qquad\qquad - \ \mathbf{PC'} : \forall t \in T, tr(\mathcal{F}) \nvdash \bar{t}.$$

Du fait que $tr(T) \subseteq tr(\mathcal{F})$, il est clair que PC'⇒ PC et PI ⇒ PI'. Il est donc naturel de se demander quelle version de ces propriétés est la plus adaptée pour qualifier un super-arbre.

Nous détaillons maintenant deux exemples qui apportent un élément de réponse à cette question et montrent que PC' est parfois trop restrictive et PI' trop permissive. En revanche, PI et PC ont le comportement escompté sur ces deux exemples. Quant à savoir s'il existe des propriétés encore plus discriminantes que PI et PC, la question reste ouverte.

Dans le premier exemple (Figure 3.3), la forêt $\mathcal{F}$ contient deux arbres sources T_1 et T_2 qui ne sont en désaccord que sur la position du taxon e. Un "super-arbre" pertinent est donc l'arbre T qui correspond à l'arbre obtenu en retirant ce taxon problématique de T_1 (ou de T_2). Cet arbre T vérifie bien PI et PC ; par contre il ne vérifie pas PC'. En effet les triplets $t = ab|c$ et $\bar{t} = ac|b$ sont tous les deux induits par $rt(\mathcal{F})$. Le premier l'est de manière triviale de par sa présence dans $rt(T_1)$, le second l'est par induction à partir des deux triplets $ac|e \in rt(T_1)$ et $ae|b \in rt(T_2)$.

Paradoxalement, PC' rejette l'arbre T en se basant sur des triplets contenant un taxon absent de T. Cet exemple se généralise facilement au cas plus fréquent où

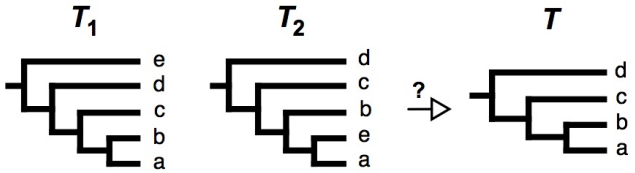

FIGURE 3.3 – **Comparaison des propriétés PC et PC'** *Une forêt $\mathcal{F} = \{T_1, T_2\}$ et un super-arbre T pouvant être proposé par une méthode de type* véto. *Ce super-arbre exclut le seul taxon problématique (e). T vérifie PI et PC, mais pas PC' ($\{ae|b, ac|e\} \vdash ac|b$).*

le super-arbre contient plus de taxons que chacun des arbres sources. Il suffit pour cela de remplacer le taxon a par $a_1a_2|a$ dans T_1 et le taxon b par $b_1b_2|b$ dans T_2 et de faire ces deux remplacements dans T (Ranwez et al., 2007a, figure 5)

Dans le second exemple (Figure 3.4), les deux arbres sources sont incompatibles, ils proposent des résolutions contradictoire $ab|c$ pour T_1 et $ac|b$ pour T_2. Dans ce cas, des clades arbitraires peuvent être validés par PI' ce qui autorise le fait de justifier des triplets du super-arbres en utilisant les triplets contradictoires. Dans cet exemple, le super-arbre T contient notamment l'information $ac|x$ qui n'est pas justifiable par les arbres sources (comme le détecte PI) mais qui peut être déduite en utilisant une des résolutions contradictoires de $\{a, b, c\}$ présentes dans les arbres sources (e.g. $\{bc|a, ab|x\} \vdash ac|x$). La formulation de PI' autorise même de justifier certains triplets de T en utilisant $ab|c$ et d'en justifier d'autres en utilisant $bc|a$.

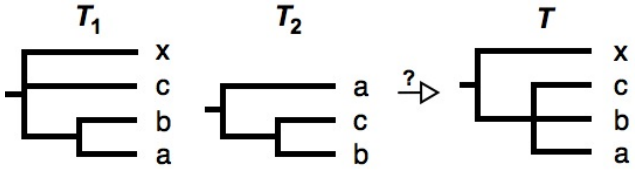

FIGURE 3.4 – **Comparaison des propriétés PI et PI'** *Une forêt $\mathcal{F} = \{T_1, T_2\}$ et un super-arbre T pouvant être proposé par une méthode de type* véto. *Ce super-arbre propose un clade contenant notamment une résolution arbitraire du triplet $ac|x$. Pourtant T vérifie PI' et PC'. En revanche T ne vérifie pas PI qui détecte le problème.*

Une autre propriété souvent mentionnée dans le cadre des super-arbres (et des consensus) stipule que le super-arbre doit contenir les informations topologiques élémentaires qui sont communes à l'ensemble des arbres sources. Si une méthode de super-arbre garantit cette propriété, on dit qu'elle est *Pareto* (Neumann, 1983). Cette propriété se décline en différentes versions suivant l'information topologique élémentaire considérée. Une méthode peut donc, par exemple, être Pareto pour les clades ou pour les triplets. Plusieurs méthodes de super-arbre sont Pareto pour les clades (Wilkinson et al., 2004). Le fait d'être Pareto sur les triplets équivaut à satisfaire la sixième propriété énoncée dans (Steel et al., 2000). Cependant cette propriété pose problème dans le cas de triplets car plusieurs arbres peuvent être

compatibles avec les arbres sources (au sens de PC ou de PC') et être Pareto. En choisir un revient à faire un choix arbitraire tandis que leur consensus strict n'est pas nécessairement Pareto. Dans ce cas, imposer le fait d'être Pareto revient donc à imposer de faire un choix arbitraire (Ranwez et al., 2007a).

3.3 *PhySIC_IST* : une méthode de super-arbre basée sur PI et PC

3.3.1 Motivations

Notre article présentant les propriétés PI et PC contient également une description d'un algorithme (*PhySIC* : PHYlogenetic Signal with Induction and non Contradiction) permettant de construire un super-arbre satisfaisant PI et PC. Cet algorithme est basé sur la décomposition en "composantes connexes" d'un graphe représentant les triplets de la forêt source. Dans ce graph, appelé *graphe de Aho* (car introduit par Aho et al., 1981), chaque sommet représente un taxon et une arête entre deux taxons a et b est présente si, et seulement si, il existe un triplet de type $ab|x$ dans la forêt source. Ainsi, lorsqu'il existe un chemin entre a et b de longueur 1 c'est qu'il existe un taxon x tel que $ab|x \in tr(\mathcal{F})$; tandis que l'existence d'un chemin plus long correspond à une série de triplets de $tr(\mathcal{F})$ qui induisent collectivement un triplet de type $ab|x$. Une *composante connexe* est constituée de l'ensemble des nœuds d'un graphe qui sont reliés par un chemin. Dans le cas du graphe de Aho, le fait d'être dans une même composante indique que les taxons doivent être dans un même clade car ils sont "vus ensembles" par d'autres taxons. On peut donc commencer la construction du super-arbre T en y intégrant ces premiers clades puis élucider leurs structures internes en considérant pour chacun un nouveau graphe de Aho qui ne prend en compte que ses taxons. On construit ainsi, de manière récursive, un super-arbre satisfaisant PC. Dans le cas où l'on observe plus de deux composantes connexes, il est possible que la création des clades correspondants engendre des relations non induites. Pour s'assurer que ce n'est pas le cas, il faut vérifier que chaque composante connexe C_i reste connexe même si l'on ne considère que le sous-ensemble de ses arêtes induites par une autre composante C_j (i.e. uniquement les arêtes correspondantes aux triplets $ab|x$ avec $x \in C_j$). *PhySIC* s'appuie sur ces propriétés du graphe de Aho pour construire un super-arbre satisfaisant PI et PC.

PhySIC a l'avantage de proposer des super-arbres parfaitement justifiés et de pouvoir indiquer la source de chacune de ses irrésolutions (i.e. si elle est due à la nécessité de satisfaire PI, PC ou les deux). Cette information permet notamment à l'utilisateur de savoir si l'ajout d'arbres sources peut permettre d'améliorer la résolution de ce clade (c'est le cas si l'irrésolution est liée à PI mais pas si elle est liée à PC). Nous avons utilisé *PhySIC* avec succès pour obtenir un super-arbre contenant 95% des genres[1] actuels de primate (Figure 3.5). Cette phylogénie des primates est largement congruente avec le "super-arbre" construit manuellement

1. le genre est un niveau taxonomique un peu plus général que l'espèce

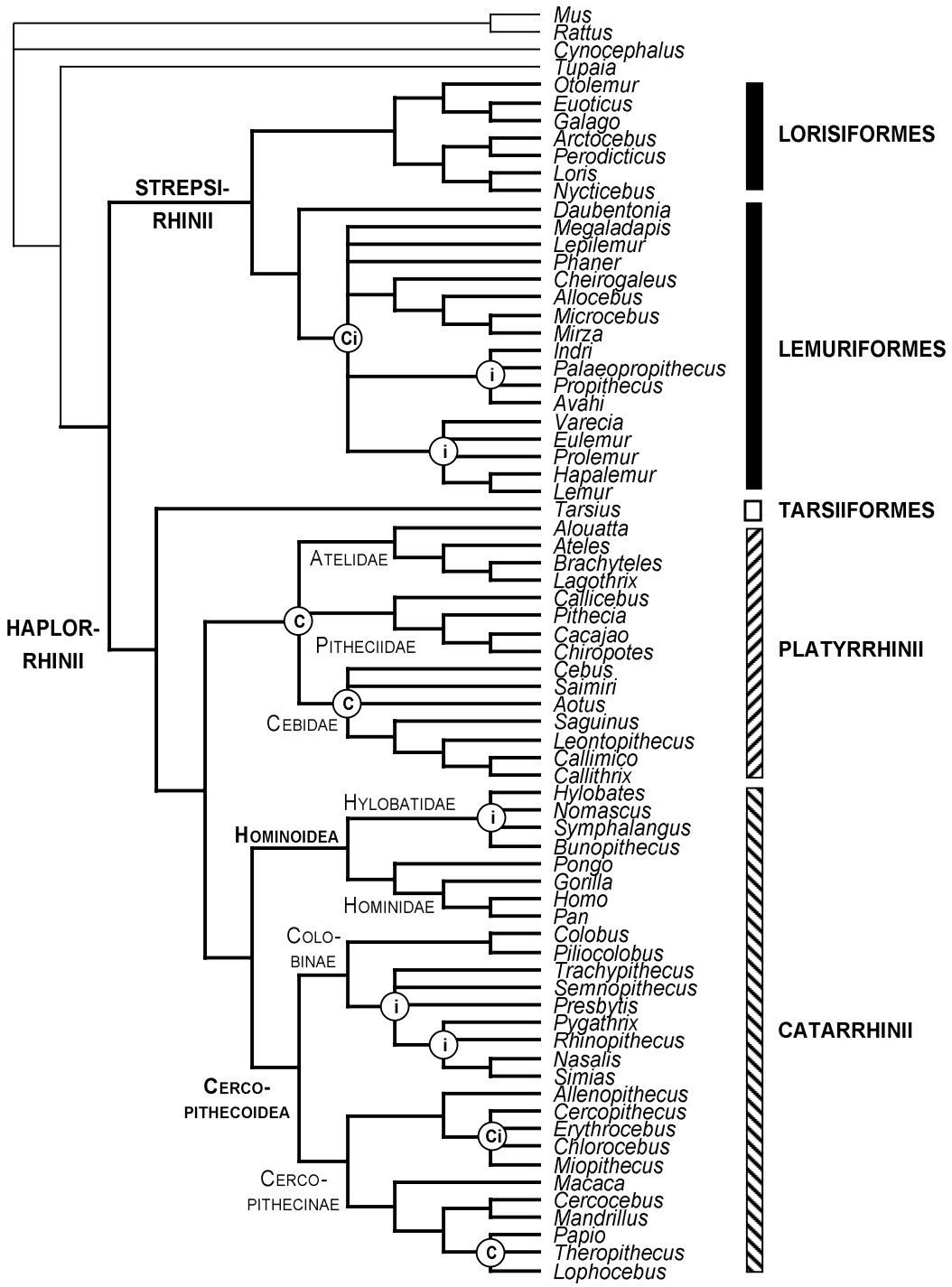

FIGURE 3.5 – **Phylogénie des primates obtenue grâce à *PhySIC*** *Ce super-arbre a été construit en appliquant* PhySIC *sur 24 arbres sources obtenus en analysants des séquences orthologues nucléaires (19 arbres) et mitochondriales (2 arbres) ainsi que des données d'événements génomiques rares (3 arbres). Les irrésolutions de ce super-arbre peuvent être liées à PC (étiquette C), à PI (étiquette i) ou aux deux (étiquette Ci).*

sur la base de l'expertise de Goodman et al. (2005). Ces résultats encourageants nous ont poussés à poursuivre ces travaux afin de surmonter les limitations dont souffraient *PhySIC*. Premièrement, le super-arbre construit par *PhySIC* contient nécessairement l'ensemble des taxons (il est *plénier*). Or dans certains cas la présence d'un seul taxon problèmatique peut obliger les méthodes de vote à proposer un *arbre complètement irrésolu* (n'ayant que les clades triviaux) alors qu'il suffit de retirer ce taxon pour pouvoir obtenir un super-arbre informatif (Figure 3.3). Deuxièmement, le critère optimisé par *PhySIC* (maximiser $\mathcal{R}_{\mathbf{T}}(\mathcal{F})$) n'est pas le plus pertinent, surtout si l'on autorise les arbres non-pléniers. Enfin, *PhySIC* souffre du même handicap que toutes les méthodes de type véto, l'ajout d'arbres sources, qui devrait améliorer la qualité du super-arbre produit, finit par la dégrader. Le problème vient du fait que, même si l'on utilise une forêt de 1 000 arbres contenant tous le triplet $ab|c$, il suffit qu'on ajoute un seul arbre contenant la résolution alternative $bc|a$ pour empêcher un super-arbre de type véto de résoudre $ab|c$. Ainsi, au delà d'un certain seuil, plus on ajoute d'arbres plus on engendre d'irrésolutions dans le super-arbre produit.

Ces limitations nous ont amenés à développer la méthode STC+*PhySIC_IST* présentée dans cette section. Cette méthode, qui est au cœur du travail de thèse de Céline Scornavacca, se démarque des méthodes de super-arbre existantes. En effet, elle intègre une première étape de vote (STC) qui élimine les informations statistiquement non représentatives des arbres sources, suivie d'une étape de véto (*PhySIC_IST*) qui combine ces arbres modifiés en un super-arbre non-plénier. Cette méthode a fait l'objet d'une publication dans la revue *BMC Bioinformatics*. Céline a implémenté STC et *PhySIC_IST* en utilisant la librairie Biopp (`http://biopp.univ-montp2.fr/`) qui facilite le développement (en C++) d'outils bio-informatiques en lien avec la biologie moléculaire et l'évolution. Julien Dutheil (CR à l'ISEM dans notre équipe) est l'initiateur et le maître d'œuvre de ce projet (Dutheil et al., 2006). J'ai également participé (de manière beaucoup plus modeste que lui) au développement de cette librairie et je fais partie des co-auteurs de l'article publié à ce sujet dans *BMC Bioinformatics*. Cette implémentation de *PhySIC_IST* (tout comme celle de *PhySIC*) est téléchargeable, mais elle peut également être utilisée en ligne à l'aide d'une interface Web évoluée (`http://www.atgc-montpellier.fr/physic_ist/`) permettant notamment de visualiser les modifications des arbres sources.

3.3.2 Principes méthodologiques de *PhySIC_IST*

PhySIC_IST est une méthode de type véto qui cherche le super-arbre le plus "informatif" respectant PI et PC. Comme nous l'avons rappelé en introduction, les méthodes de type véto "éliminent" les conflits en introduisant des irrésolutions ou en écartant des taxons. *PhySIC_IST* est la première méthode de véto à considérer simultanément ces deux possibilités. Elle peut donc, dans un même super-arbre choisir tantôt d'introduire une irrésolution tantôt d'écarter un taxon problématique afin d'obtenir globalement la solution la plus "informative". Après avoir défini cette

notion "d'informativité", nous décrirons les principes heuristiques utilisés dans *PhySIC_IST*, et nous terminerons cette section sur la description du pré-traitement STC qui, couplé à *PhySIC_IST* donne de très bons résultats.

3.3.2.1 Comparer "l'informativité" d'arbres n'ayant pas le même nombre de feuilles

L'objectif de *PhySIC_IST* n'est pas seulement de trouver un arbre (plénier ou non) satisfaisant PI et PC, mais de trouver l'arbre le plus informatif satisfaisant ces deux propriétés. En effet il existe potentiellement de nombreux arbres satisfaisant PI et PC, dont certains, n'ont que peu (ou pas) d'intérêt (e.g. l'arbre complètement irrésolu). Il est donc nécessaire de pouvoir comparer la quantité d'information phylogénétique (i.e. *l'informativité*) de différents arbres. Dans la suite, nous noterons $L(T)$ l'ensemble des feuilles d'un arbre T. Cette notation se généralise naturellement à une forêt de la manière suivante : $L(\mathcal{F}) = \bigcup_{T_i \in \mathcal{F}} L(T_i)$.

Pour évaluer l'informativité d'un arbre T on peut se baser sur le nombre de triplets que contient cet arbre (i.e. $|tr(T)|$) ou utiliser une mesure plus complexe, issue de la théorie de l'information et appelée *CIC* (pour Cladistic Information Content). L'utilisation du CIC a été proposée par Thorley et al. (1998) dans le cas du consensus. Nous avons généralisé leur définition afin de pouvoir également l'appliquer à des super-arbres non pléniers. Dans tous les cas, un arbre est d'autant plus informatif qu'il limite le nombre de scénarios évolutifs possibles pour les n espèces étudiées, i.e. $n = L(\mathcal{F})$. Le CIC d'un arbre T est donc défini en comparant le nombre $n_R(n)$ de scénarios évolutifs complètement résolus portant sur ces n espèces avec le nombre $n_R(T, n)$ de ces scénarios qui sont compatibles avec T. Plus précisément,

$$CIC(T, n) = -\log\Big(n_R(T, n)/n_R(n)\Big) \tag{3.1}$$

Afin de comparer les résolutions de super-arbres obtenus pour des forêts de tailles différentes, il est nécessaire de normaliser cette valeur :

$$CIC_N(T, n) = CIC(T, n)/(-\lg 1/n_R(n)). \tag{3.2}$$

Dans le cas du consensus, ou de super-arbre plénier, la valeur $n_R(T, n)$ ne dépend que des irrésolutions présentes dans T (qui peuvent être résolues de plusieurs manières et sont donc compatibles avec plusieurs histoires évolutives). Pour gérer des super-arbres non pléniers, il faut également prendre en compte les taxons présents dans la forêt source mais absents de T (sur lesquels T n'apporte donc aucune information). Bien que cela complique le calcul de $n_R(T, n)$, nous avons proposé un algorithme linéaire (i.e. en $O(n)$) permettant de calculer $CIC(T, n)$ (il peut facilement être adapté pour calculer $CIC_N(T, n)$). Cet algorithme s'appuie sur le fait, que pour un nœud interne u_i ayant c_i fils, il existe $(2c_i - 3)!!$[2] résolutions binaires distinctes (Semple and Steel, 2003). Les résolutions des nœuds étant indépendantes les unes des autres, on peut facilement calculer le nombre n_r d'arbre complètement

2. n!! = n *(n-2) * (n-4)...

résolus ayant les mêmes feuilles que T. Le nombre de scénarios possibles résultants de l'ajout des taxons absents de T est le même pour chacun de ces arbres. On peut déterminer ce nombre en observant qu'un taxon manquant peut être inséré sur n'importe laquelle des $2 * |L(T)| - 1$ branches d'un tel arbre. Cette insertion crée alors deux nouvelles branches et il y a donc $2 * (|L(T)| + 2) - 1$ manières d'insérer le taxon suivant, etc. Ces remarques conduisent à la solution linéaire décrite dans l'algorithme 3 (N.B. l'algorithme est bien linéaire, malgré la présence de deux boucles imbriquées, car ces deux boucles n'engendrent au total qu'une multiplication par branche de l'arbre).

Algorithm 3: Calcul de l'informativité d'un arbre (CIC)

Data: Un arbre T, le nombre n de feuilles considérées et une fonction $(n_R(n))$ qui renvoie le nombre d'arbres binaires à n feuilles.
Result: Algorithm $CIC\,(T, n)$.
$nr_{T,n} \leftarrow 1$;
$I \leftarrow$ l'ensemble des nœuds internes de T ;
foreach $u \in I$ **do**
    $c \leftarrow |children(u)|$;
    for j *in* $[2, c]$ **do**
        $nr_{T,n} \leftarrow (nr_{T,n} * (2 * j - 3))$;
$max \leftarrow n - |L(T)|$; $j \leftarrow |L(T)|$;
for k *in* $[1, max]$ **do**
    $nr_{T,n} \leftarrow (nr_{T,n} * (2 * j - 1))$;
    $j \leftarrow j + 1$;
$nr_n \leftarrow (2n - 1)!!$
return $-\log(nr_{T,n}/n_R(n))$

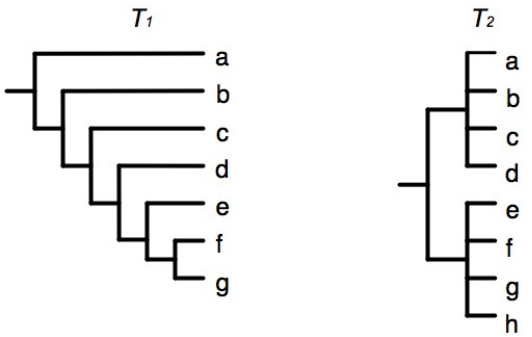

FIGURE 3.6 – **Comparaisons de deux mesures d'informativité : CIC et nombre de triplets.** *T_1 et T_2 sont deux super-arbres possibles pour une forêt ayant $\{a, b, c, d, e, f, g, h\}$ comme feuilles. Le second arbre contient plus de triplets que le premier ($|\mathcal{R}(T_1)| = 35$ alors que $|\mathcal{R}(T_2)| = 48$). Cependant,le premier arbre, bien que pénalisé par l'absence de h obtient néanmoins un meilleur CIC car il est nettement plus résolu ($\mathrm{CIC}_N(T_1, 8) = 0.78$ alors que $\mathrm{CIC}_N(T_2, 8) = 0.54$).*

Il n'est donc pas très coûteux (en temps) de calculer $CIC(T, n)$. On peut néanmoins se demander si cette mesure est réellement plus pertinente que la simple mesure du nombre de triplets de T. La mesure de $CIC(T, n)$ a évidemment l'avantage de s'appuyer sur les fondements solides de la théorie de l'information et de

refléter une notion concrète pour le biologiste (combien de scénarios évolutifs sont compatibles avec le super-arbre). De plus, cet avantage se retrouve dans la capacité du CIC à proposer une hiérarchisation de l'informativité des arbres qui semble plus en accord avec la vision des biologistes comme l'illustre l'exemple de la Figure 3.6. Dans cet exemple, il est clair que l'arbre T_1 favorisé par la mesure de CIC (car l'exclusion du taxon h lui permet de résoudre complètement la phylogénie des autres taxons) est aussi celui qui aura la préférence des biologistes par rapport à l'arbre T_2 contenant tous les taxons (et plus de triplets) mais qui est très peu résolu.

3.3.2.2 Principe algorithmique de l'heuristique *PhySIC_IST*

L'objectif de *PhySIC_IST* est de trouver l'arbre le plus informatif satisfaisant PI et PC. Ce qui correspond à résoudre le problème d'optimisation suivant :

Problème	Most Informative Induced and non-Contradicting Supertree (MIICS)
Données	une collection $\mathcal{F}$ d'arbres enracinés.
Résultats	un arbre T tel que : (i) T satisfait PI and PC for $\mathcal{F}$ (ii) $CIC(T, \lvert L(\mathcal{F})\rvert)$ est maximum parmi les arbres vérifiant (i).

Ce problème est une variante des problèmes MIST (Maximum Identifying Substet of rooted Triplets) et ST (Triplet Supertree) qui sont tous les deux NP-complets (Bryant, 1997; Guillemot and Berry, 2007). Le lien avec le problème MIST n'est pas le fait du hasard, puisque la preuve de NP-complétude de MIST a été faite dans le cadre du travail de l'article (Ranwez et al., 2007a) décrivant PI, PC et *PhySIC*. Cette preuve n'a finalement pas été intégrée à l'article afin de ne pas l'alourdir et n'a finalement fait l'objet que d'un rapport interne du LIRMM. Bien que nous n'ayons pas, cette fois, rédigé de preuve formelle, la similitude de ces problèmes avec MIICS rend la conjecture de sa NP-complétude raisonnable. *PhySIC_IST* est donc une heuristique pour résoudre ce problème. Il est cependant important de noter que l'aspect heuristique de *PhySIC_IST* ne porte que sur le point (ii) et que l'arbre qu'il renvoie vérifie toujours PI et PC.

PhySIC_IST utilise une heuristique gloutonne (i.e. les choix d'une étape ne sont jamais remis en cause par la suite) et construit le super-arbre par insertion itérative des différents taxons. L'ordre d'insertion est défini de manière à traiter en premier les taxons qui sont, a priori, les plus simples à insérer. Les premiers taxons considérés sont donc ceux pour lesquels on dispose d'un maximum d'informations (i.e. ils sont présents dans un maximum de triplets de $tr(\mathcal{F})$) et qui sont impliqués dans un minimum de conflits (i.e. ils sont présents dans un minimum de triplets t tels que $t \in tr(\mathcal{F})$ et $\bar{t} \in tr(\mathcal{F})$). Lorsque l'on cherche à insérer un taxon t sur l'arbre courant T, on affecte une valeur de support à chaque branche et à chaque nœud de T. Ce support correspond au nombre d'arbres de la forêt qui sont compatibles avec l'histoire évolutive qui sera obtenue si t est "greffé" à cet endroit. La Figure 3.7 indique l'ensemble des zones de l'arbre courant T qui sont supportées par un arbre source

T_i pour insérer le taxon z. Dans un premier temps, un taxon n'est greffé que s'il

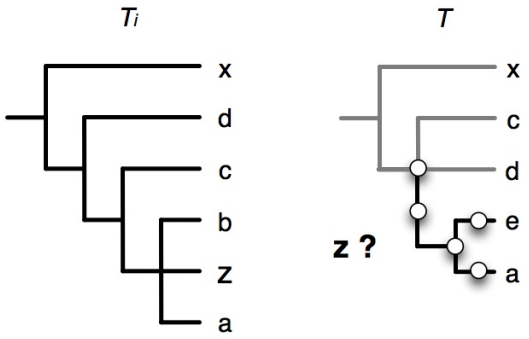

FIGURE 3.7 – **Définition des valeurs de support dans l'heuristique *PhySIC_IST*.** *Lors de l'insertion du taxon z dans l'arbre courant T, l'arbre source T_i définit une zone connexe (en gras) de l'arbre T dans laquelle z peut être greffé sans créer de conflit avec T_i. Cette zone contient cinq endroits possibles (les cercles blancs) pour greffer z en accord avec T_i, ces cinq endroits vont donc voir leur support augmenter de 1.*

existe une position unique ayant le support de tous les arbres sources. Cette condition garantit que l'arbre construit satisfait PC (du fait du support de l'ensemble des arbres) et PI (du fait de l'unicité de cette position). Si une telle position n'existe pas, on passe au taxon suivant. Le taxon non greffé, sera re-considéré ultérieurement car l'ajout de taxons supplémentaires peut aider à le positionner correctement. Quand il n'est plus possible d'insérer de taxons, les contraintes sont peu à peu relâchées. L'insertion d'un taxon t dans un arbre T satisfaisant PI et PC peut alors produire un arbre T' ne satisfaisant plus ces propriétés. Des branches de T' sont alors écrasées (en utilisant des procédures développées pour *PhySIC*) de manière à obtenir un arbre T'' satisfaisant PC et PI. On compare alors l'informativité de T et de T''. Si $CIC(T'', L(\mathcal{F})) > CIC(T, L(\mathcal{F}))$ alors T'' devient l'arbre courant, sinon T reste l'arbre courant et l'on teste l'insertion du taxon suivant.

Comme nous avons vu qu'il suffisait de tester la présence de contradictions directe pour assurer PC, l'algorithme *PhySIC_IST* s'appuie sur deux ensembles de triplets, l'un (noté $\mathcal{R}$) contient les triplets des arbres sources, et l'autre (noté $\mathcal{R}_{dc}$) est le sous-ensemble de ces triplets qui sont en contradiction directe dans les arbres sources. L'ensemble $\mathcal{R}$ permet de tester la validité de PI tandis que l'ensemble $\mathcal{R}_{dc}$ suffit à tester celle de PC. Cependant, *PhySIC_IST* exploite également la structure des arbres sources pour affecter les supports des branches de l'arbre courant sans avoir à considérer l'ensemble des triplets. Les différentes étapes de *PhySIC_IST* sont détaillées sous forme de pseudo-code dans (Scornavacca et al., 2008) et plus encore dans (Scornavacca, 2009, chapitre 4). Ces publications incluent également la preuve que le super-arbre produit satisfait bien PI et PC et que la complexité en temps de *PhySIC_IST* est de $O(n^3(k+n^3))$ avec $n = L(\mathcal{F})$ et $k = |\mathcal{F}|$.

3.3.2.3 Détecter et éliminer des erreurs présentes dans les arbres sources

Comme nous l'avons évoqué en préambule, l'ajout de nombreux arbres sources peut nuire à la qualité de l'arbre produit par une méthode de type véto. On peut limiter ce problème en essayant de s'assurer de la fiabilité des clades de chaque arbre et en éliminant les clades les moins fiables. Une manière d'estimer la fiabilité des clades d'un arbre T est basée sur le ré-échantillonage des données utilisées pour l'inférer. On peut, par exemple, produire 100 variantes de l'alignement de séquences (en ré-échantillonant ses sites), inférer un arbre à partir de chacun de ces alignements et utiliser la fréquence d'apparition d'un clade dans cette collection comme mesure de sa fiabilité. On associe ainsi une valeur de support (appelée *bootstrap*) à chacun des clades de l'arbre T. Il est courant de ne conserver que les clades de T qui dépassent un certain seuil (les seuils de 50, 70 et 90% sont souvent utilisés [3]). Si l'on utilise des méthodes de véto pour combiner des arbres de gènes, et non pas des phylogénies fiables publiées, il est important de s'assurer de la fiabilité des informations que l'on utilise en ne conservant que les clades les plus fiables de chaque arbre.

L'utilisation des valeurs de bootstrap pour mesurer la fiabilité d'une information traite chaque arbre individuellement. La confrontation des différences observées entre ces arbres apporte aussi un éclairage sur la fiabilité de l'information présente dans chaque arbre source. Si par exemple on considère une forêt de 100 arbres sources dans laquelle on observe 48 fois la résolution $ab|c$ et 52 fois $ac|b$, il semble effectivement prudent de ne pas résoudre ce triplet dans le super-arbre proposé. Si par contre on observe 99 fois $bd|a$ et une seule fois la résolution alternative $ba|d$ (dans un arbre T_{bad}), il semble plus judicieux d'autoriser la présence du triplet $bd|a$ dans le super-arbre proposé et d'ignorer la résolution $ba|d$ qui ne reflète très certainement pas l'histoire des espèces. Pour cela, nous utilisons un test statistique (basé sur le χ^2) qui nous permet de détecter les triplets (t) qui sont significativement moins fréquents que leurs alternatives ($\bar{t}$) et que l'on peut ignorer sans (trop de) risque (le seuil du test est un paramètre de cette phase de pré-traitement).

De plus, cette information sur l'aspect improbable d'un triplet $ba|d$ nous indique également que les clades de l'arbre T_{bad} qui engendrent ce triplet ne correspondent probablement pas à une information fiable (sur l'histoire des espèces) même si ils ont une valeur de bootstrap élevée. Tout comme on peut écraser les arêtes de T_{bad} ayant un faible bootstrap, on peut également écraser celles conduisant à l'observation du triplet aberrant $ba|d$. Ceci dit, il est possible que cela conduise à écraser de nombreuses arêtes alors que le problème vient peut-être uniquement d'un taxon mal placé qu'il suffirait d'enlever. La question est alors de savoir quelle correction donne la variante de T_{bad} la plus informative. Pour cela, on peut utiliser *PhySIC_IST* en lui fournissant comme forêt $\{T_{bad}\}$ et comme ensemble $\mathcal{R}_{dc}$ l'ensemble des triplets aberrants détectés. L'arbre ainsi obtenu ne contient alors que des informations qui étaient présentes dans T_{bad} et qui sont plausibles au vu des autres arbres. D'une certaine manière, ce pré-traitement "corrige" donc les arbres sources, d'où sont nom

3. tout comme le seuil de 5% pour la p-value, ces seuils sont plus des normes de fait que des choix justifiables scientifiquement.

de STC pour *Source Tree Correction*.

On peut utiliser ce pré-traitement sur les arbres sources, dont les clades de faible valeur de bootstrap ont (ou non) déjà été enlevés, ou l'appliquer sur la forêt contenant l'ensemble des réplicats de chaque arbre. Cette dernière solution n'est évidemment envisageable que lorsqu'on dispose des données initiales ayant permis d'inférer les arbres sources, ce qui n'est pas toujours le cas.

3.3.3 Validation et application à la phylogénie des Triticeae

Afin de valider notre approche, nous avons comparé les performances de *PhySIC* et *PhySIC_IST* (avec et sans le pré-traitement STC) à celles obtenues avec MRP sur des jeux de données simulés. Parralèlement, nous avons utilisé *PhySIC_IST* pour élucider les relations phylogénétiques au sein des *Triticeae* (le clade contenant des céréales telles que le blé et l'orge).

3.3.3.1 Validation sur des données simulées

Nous avons utilisé un protocole de simulation classique dans le cadre de l'évaluation des méthodes de super-arbre. Ce protocole proposé par Eulenstein et al. (2004) puis repris et adapté par Criscuolo et al. (2006) est représenté de manière schématique dans la Figure 3.8. Ce protocole permet de générer des alignements de séquences correspondant à des gènes ayant pu évoluer à des vitesses différentes au cours du temps. Pour chacun de ces alignements, on ne conserve qu'une partie des séquences afin de mimer le fait que des gènes peuvent avoir été perdus, ou ne pas avoir été séquencés, chez certaines espèces. A partir de ces alignements on infère un arbre de gènes (suivant le principe de maximum de vraisemblance) dont on ne garde que les clades les plus fiables (bootstrap >50%). On utilise alors plusieurs de ces arbres de gènes pour inférer un super-arbre que l'on peut ensuite comparer à l'arbre des espèces utilisé pour démarrer la simulation. La qualité des super-arbres inférés, varie en fonction de deux paramètres importants de ces simulations. Le premier est le paramètre d (pour *deletion*) qui représente le pourcentage de séquences enlevées de l'alignment. Ce pourcentage peut être le même pour tous les gènes d'une forêt et valoir 25%, 50% ou 75% ou bien, ce qui est plus réaliste, il peut varier d'un gène à l'autre ($d = mix$). Plus ce pourcentage est élevé, moins il y a de taxons dans les arbres de la forêt. Chaque arbre contient donc moins d'informations et les recoupements entre ces informations sont plus faibles. La qualité des super-arbres inférés diminue donc globalement lorsque la variable d augmente. Le second facteur clef, est le nombre d'arbres présents dans la forêt. Plus ce nombre k est élevé plus on dispose d'information et il semble donc logique que la qualité des super-arbres augmente avec cette valeur. Pour chaque couple de valeur (d, k) 100 jeux de données sont simulés et l'on évalue la performance d'une méthode pour cette condition en prenant la valeur moyenne sur ces 100 réplicats. Plusieurs indicateurs de qualités sont analysés dans (Scornavacca, 2009) ; nous nous focaliserons ici sur deux indicateurs principaux. Le premier est l'informativité du super-arbre – mesurée par son

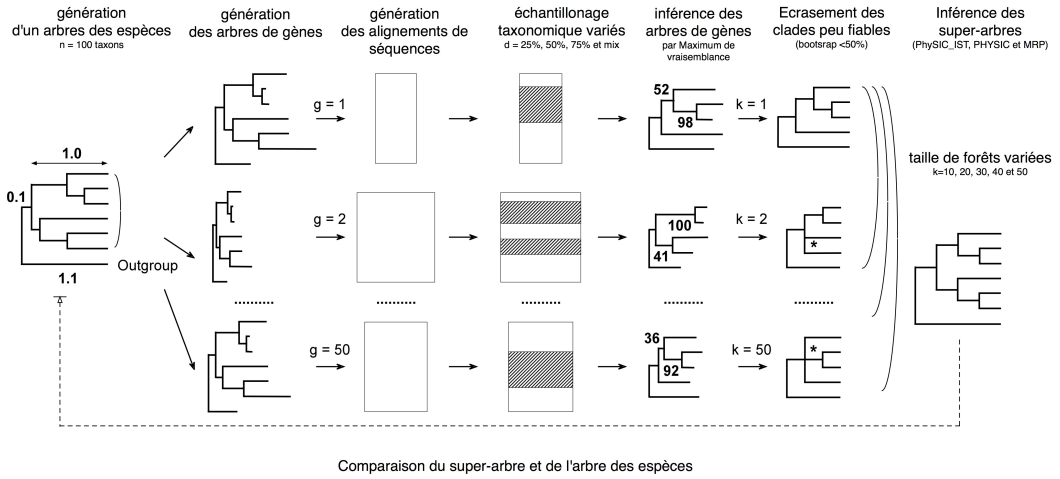

FIGURE 3.8 – **Protocole de simulation** *Ce schéma résume le protocole que nous avons utilisé pour simuler des forêts contenant un nombre d'arbres de gènes (k) et un échantillonnage taxonomique (d) variables. Plus de détails sont fournis dans le corps du texte.*

CIC_N (Figure 3.9)– et le second est le pourcentage d'informations erronées qu'il contient – mesuré par le pourcentage de triplets qui sont dans ce super-arbre mais pas dans l'arbre des espèces (Figure 3.10).

Un bon super-arbre doit évidemment être le plus informatif possible tout en contenant un minimum d'erreurs. A ce titre, *PhySIC_IST* constitue une amélioration significative de *PhySIC*, et ce quelles que soient les valeurs de d et de k. L'apport du pré-traitement STC est aussi remarquable. Il permet d'améliorer de manière nette l'informativité des arbres proposés par *PhySIC* et *PhySIC_IST* en modifiant très peu le pourcentage d'erreurs qu'ils contiennent. Le gain lié à STC est particulièrement net dans le cas $d = 25\%$ et $k = 50$. Ce qui n'est pas surprenant car cela correspond au cas où l'on dispose du maximum de triplets dans la forêt d'arbres sources. Cette augmentation de la quantité d'information disponible, qui est un handicap potentiel pour les méthodes de véto, est ici compensée par l'utilisation du STC. En effet, elle permet d'observer de nombreuses résolutions d'un même triplet dans des arbres différents ; le STC peut alors détecter, et éliminer, les résolutions qui sont significativement minoritaires. A l'inverse, pour la condition $d = 75\%$ et $k = 10$, l'apport du pré-traitement est quasiment nul car il y a tellement peu d'information dans la forêt source que les différences de fréquence entre des triplets alternatifs ne sont presque jamais significatives.

Ces simulations montrent également que l'utilisation couplée de STC et de *PhySIC_IST* (STC+*PhySIC_IST*) produit des arbres quasiment aussi informatifs que ceux proposés par MRP mais contenant beaucoup moins d'erreurs. L'utilisation de STC+*PhySIC_IST* semble donc bien constituer une solution de choix pour assem-

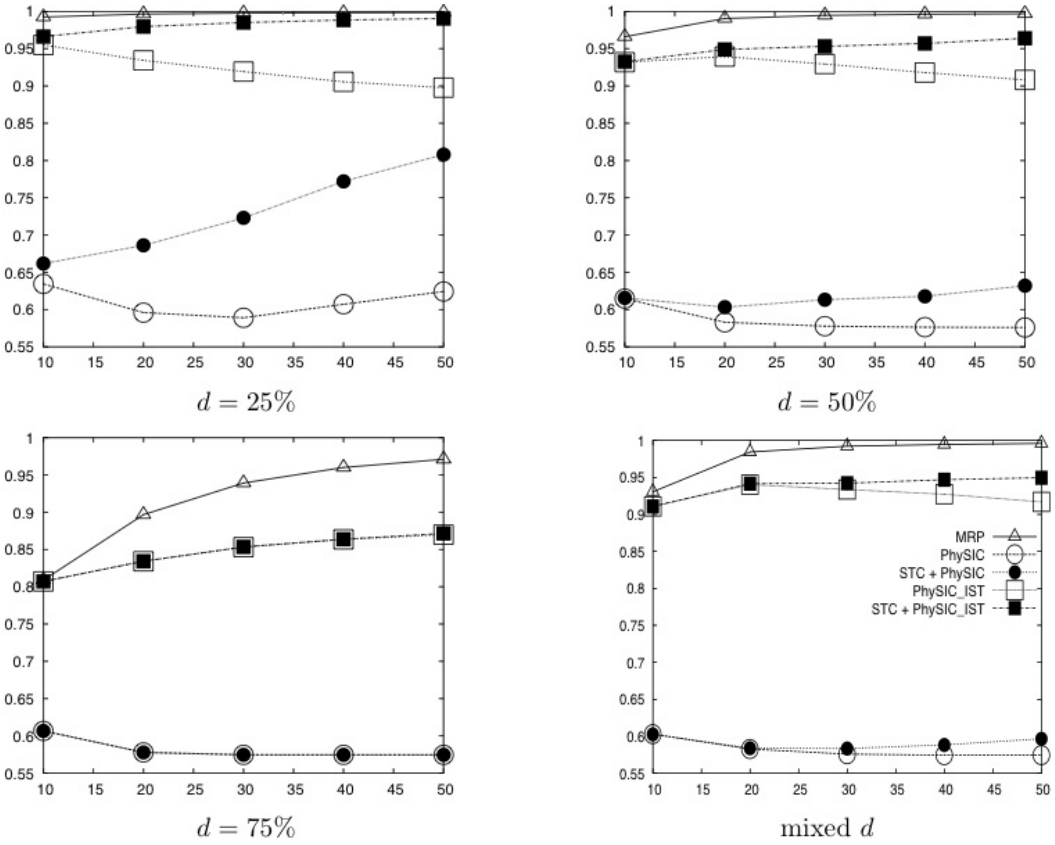

FIGURE 3.9 – **Informativité des super-arbres (CIC_N)** *Valeurs moyennes du* CIC_N *(sur 100 réplicats) des super-arbres inférés par MRP* △*,* PhySIC ○*,* PhySIC_IST □*, STC+*PhySIC ● *and STC+*PhySIC_IST ■*), en fonction du nombre d'arbres sources (*$k = 10,\ 20,\ 30,\ 40$ *et* 50*). Ces résultats sont fournis pour des arbres sources ayant un pourcentage de taxons manquants par arbre de d=25%, 50%, 75% ou un pourcentage variable d'un arbre à l'autre (*$d = mix$*)*

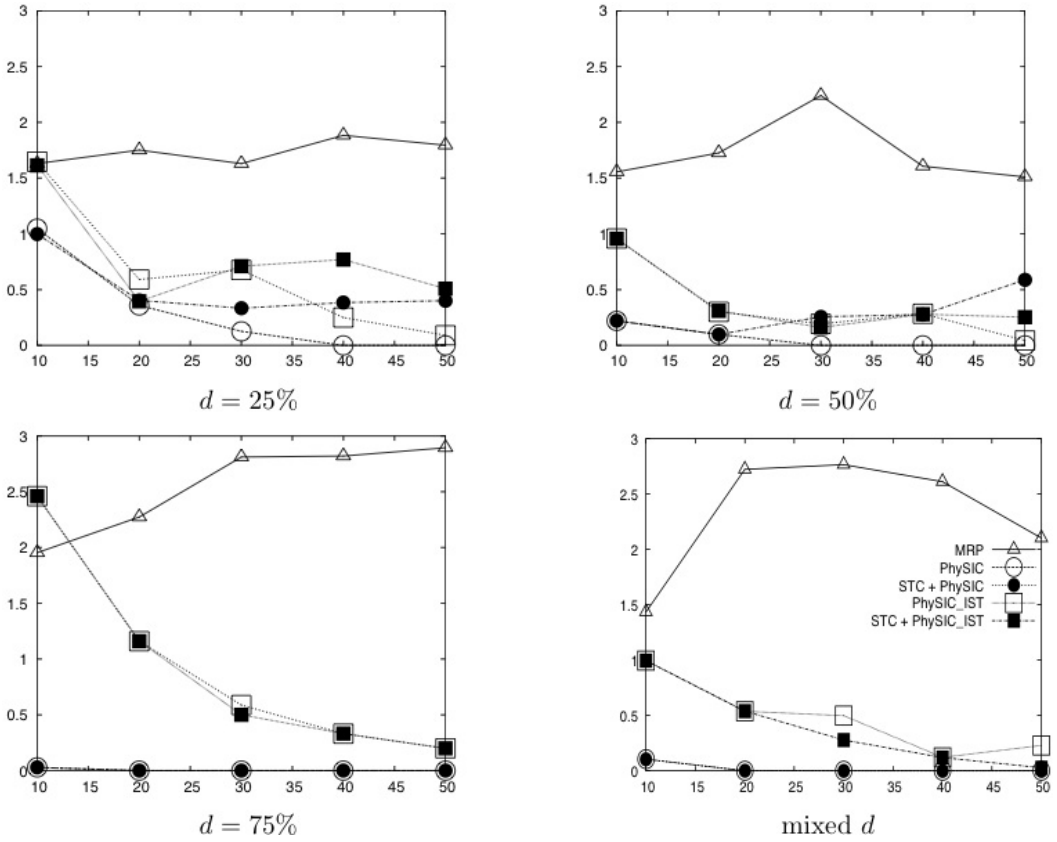

FIGURE 3.10 – **Pourcentage de triplets erronés dans les super-arbres (ou erreur de type I)** *Pourcentages moyens de triplets erronés (sur 100 réplicats) pour MRP* △*,* PhySIC ○*,* PhySIC_IST □*, STC+*PhySIC ● *et STC+*PhySIC_IST ■*, en fonction du nombre d'arbres sources (*$k = 10,\ 20,\ 30,\ 40$ *et* 50*). Ces résultats sont fournis pour des arbres sources ayant un pourcentage de taxons manquants par arbre de : d=25%, 50%, 75% ou un pourcentage variable d'un arbre à l'autre (*$d = mix$*)*

bler des phylogénies partielles en vue d'obtenir une phylogénie plus large qui soit à la fois fiable et informative. Ceci est également confirmé par les bons résultats obtenus sur des jeux de données réels. *PhySIC_IST* nous a notamment permis de combiner avec succès les arbres de 94 marqueurs protéiques pour produire un super-arbre contenant 79 représentants du règne animal (Scornavacca et al., 2008, Figure 9).

3.3.3.2 Un nouvel éclairage sur la phylogénie des Triticeae

Les *Triticeae* sont un groupe [4] taxonomique de plantes qui contient notamment le blé (*wheat*), le seigle (*rye*) et l'orge (*barley*). L'étude de ces céréales recouvre des enjeux alimentaires (et économiques) importants. Les résultats résumés dans cette section sont le fruit d'une collaboration étroite entre l'ISEM, l'INRA (l'Institut National de la Recherche Agronomique) et SupAgro (Centre international d'études supérieures en sciences agronomiques). Ils font l'objet d'une publication soumise à la revue *Systematic Biology* (Escobar et al., soumis) dont Céline et moi sommes co-auteurs.

Bien que de nombreuses études phylogénétiques aient été récemment menées sur ce groupe leurs conclusions contradictoires ne permettent pas de dégager une histoire évolutive claire. La plupart de ces études se base sur des gènes nucléaires présents en une seule copie (Mason-Gamer, 2001; Petersen and Seberg, 2002; Helfgott and Mason-Gamer, 2004; Mason-Gamer, 2005) ou sur l'analyse de gènes chloroplastiques (Petersen and Seberg, 1997; Mason-Gamer et al., 2002; Yamane and Kawahara, 2005). Leurs résultats sont si contradictoires que quasiment aucun clade consensuel ne s'en dégage. Cependant, chacune de ces études ne se base que sur un nombre très faible de gènes, en fait la plupart n'en utilise même qu'un seul. Or, comme nous l'avons vu dans la section 1.3.2, plusieurs raisons biologiques et méthodologiques peuvent amener l'histoire d'un gène à différer de celle des espèces.

Un effort important de séquençage a permis à nos partenaires de constituer un jeu de données incluant des séquences issues de 27 gènes (26 nucléaires et un chloroplastique) et couvrant 19 espèces de *Triticeae*. En analysant ces données nous avons pu non seulement confirmer la diversité des histoires évolutives de ces gènes, mais également mettre en évidence le signal phylogénétique qui en émerge. A partir de ces 27 marqueurs phylogénétiques, nous avons conduit à la fois une analyse de type super-matrice et une analyse de type super-arbre (en utilisant STC+*PhySIC_IST*). Ces deux analyses s'accordent pour identifier un certain nombre de constantes dans le signal phylogénétique de ces gènes. La figure Figure 3.11 présente les clades mis en évidence par ces analyses. Le fait que ces clades, numérotés de I à V, soient identifiés par ces deux analyses renforce la confiance qu'on peut avoir dans leur validité.

Afin de mieux comprendre l'hétérogénéité des arbres de gènes, nous avons étudié les conflits qui existent entre ces arbres et la phylogénie des espèces (inférée par l'approche super-matrice). Dans de nombreux cas, ces conflits ne sont pas dus à une histoire évolutive différente du gène mais à une résolution quasi aléatoire de certaines parties de son arbre (causée par le manque de signal phylogénétique de

4. de manière taxonomiquement plus précise, il s'agit d'une *tribu*.

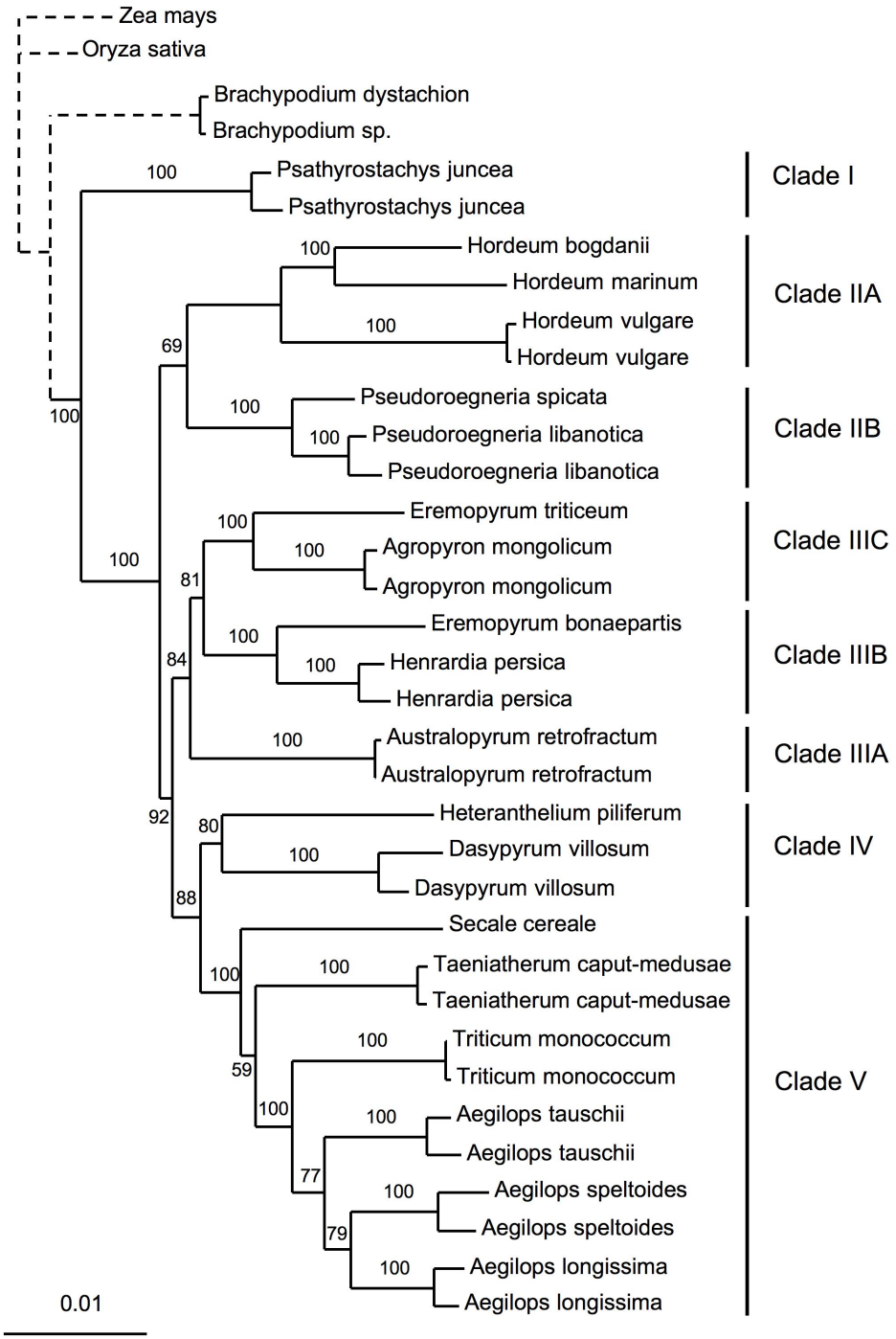

FIGURE 3.11 – **Phylogénie des Triticeae obtenue par une analyse de type super-matrice** *Cette phylogénie a été obtenue en utilisant le logiciel PAUP Swofford (2003) pour chercher l'arbre le plus vraisemblable selon un modèle d'évolution des séquences de type GTR et en modélisant l'hétérogénéité des vitesses d'évolution des différents sites en utilisant une loi gamma (et une catégorie de sites invariants). Les valeurs indiquées à côté de chaque nœud sont les valeurs de bootstrap. Les branches en pointillées sont réduites d'un facteur 10.*

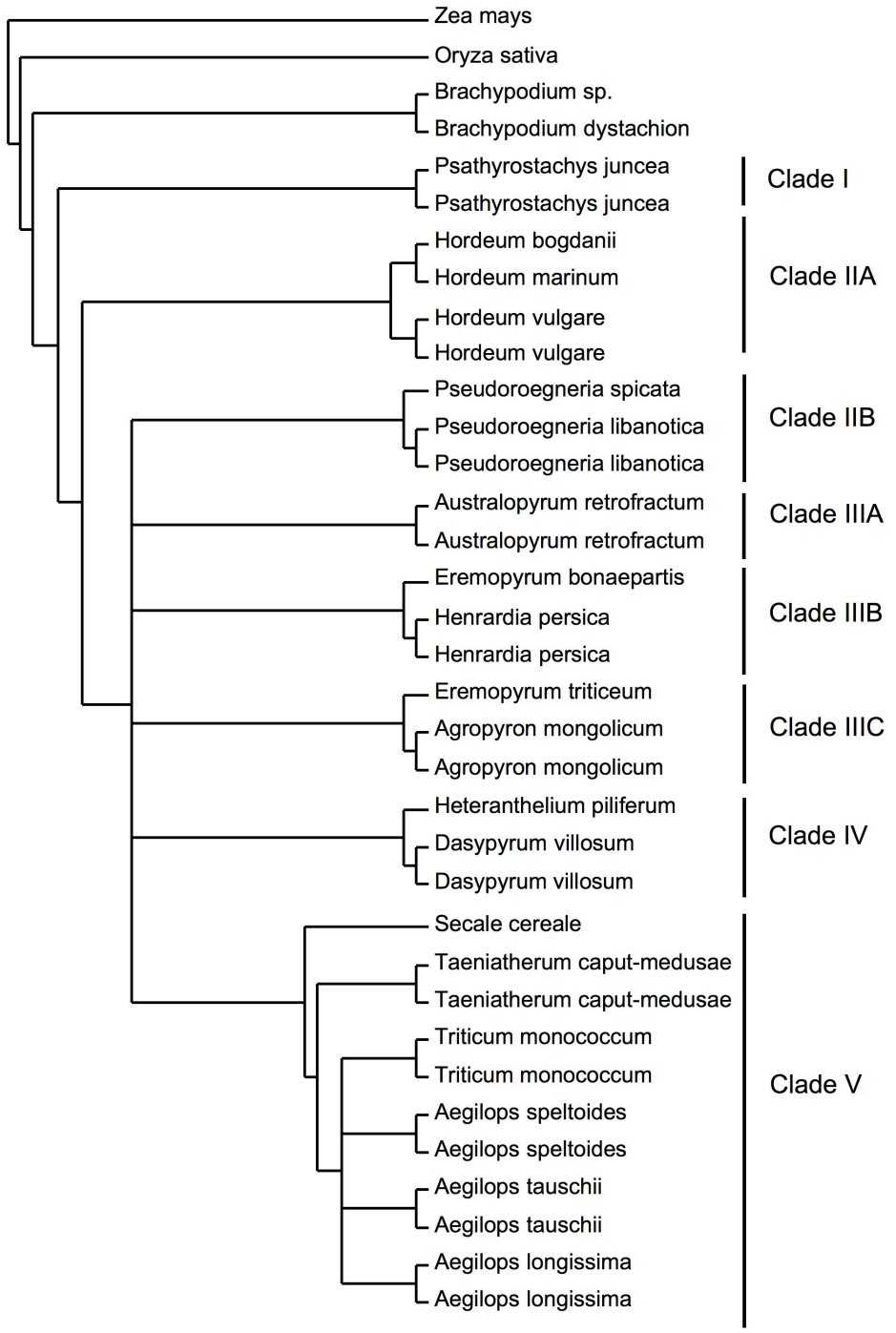

FIGURE 3.12 – **Phylogénie des Triticeae obtenue par STC+*PhySIC_IST***
*Cette phylogénie a été obtenue en utilisant STC+*PhySIC_IST *pour analyser la forêt constituée des 100 réplicats de bootstrat de chacun des 27 gènes (i.e. 27 000 arbres sources).*

ce marqueur). Pour différencier ces conflits apparents des conflits réels, nous avons appliqué une variante du test statistique de notre pré-traitement STC sur les 100 réplicats de bootstrap de chaque arbre. Nous avons ainsi pu quantifier le nombre de conflits significatifs entre deux gènes ainsi qu'entre un gène et la super-matrice.

L'analyse de ces données nous a permis de mettre en évidence une corrélation entre le nombre de conflits qui existent entre deux gènes et leur proximité sur le chromosome. Une telle corrélation avait déjà été observée par Pollard et al. (2006) dans le génome de la mouche (*Drosophila*) tandis que la répartition des zones de conflit semble aléatoire (Zou et al., 2008) dans le génome de *Oryza* (une espèce de riz). Un autre résultat émerge de l'analyse de ces données. Parmi les 21 gènes de notre analyse qui sont sur le chromosome 3, ceux qui sont les plus proches du centromère sont ceux ayant l'histoire évolutive la plus congruente avec la phylogénie des espèces. Nous avons évidemment vérifié que ce résultat n'était pas simplement dû à une sur-représentation des gènes centromériques dans les données utilisées pour inférer l'arbre des espèces. Une explication possible est liée au fait que le nombre de recombinaisons (qui brouillent l'histoire évolutive) est généralement plus faible près du centromère d'un chromosome. Quoi qu'il en soit, ces deux corrélations, sont des indicateurs forts du fait que les hétérogénéités entre arbres de gènes ont des causes biologiques réelles et ne sont pas uniquement dus à des biais méthodologiques.

Cette analyse, exploitant le STC et *PhySIC_IST*, confirme donc que les Triciteae ont une histoire évolutive complexe tout en permettant de dégager des clades fiables au sein de ce groupe. D'autre part, en regardant de plus près les séquences exclues des arbres de gènes par le pré-traitement STC, nous avons découvert une erreur sur l'une d'entre elles. Cette anecdote illustre l'un des intérêt qui existent à considérer les arbres de gènes individuellement et à utiliser une approche autorisant le retrait de taxons.

3.4 PhyloExplorer : un outil pour gérer et analyser une collection d'arbres évolutifs

3.4.1 Les origines du projet

Comme l'illustrent les sections précédentes, une part importante de mes travaux porte sur les super-arbres. La première étape de ce type d'analyses consiste à collecter et préparer une forêt d'arbres sources qui soit pertinente pour la question phylogénétique posée. Par exemple, si l'on s'intéresse à l'histoire évolutive des mammifères, on peut commencer par chercher parmi les milliers de phylogénies publiées et recensées dans TREEBASE (Sanderson et al., 1994), celles qui semblent pertinentes (e.g. qui contiennent au moins six mammifères). Même pour un taxonomiste aguerri, cette tâche est fastidieuse. Pour moi, elle est quasiment impossible. Il faut ensuite éliminer les taxons inutiles de ces arbres, s'assurer que les noms utilisés pour chaque taxon sont cohérents d'un arbre à l'autre, calculer les statistiques permettant d'évaluer la pertinence globale de la forêt obtenue (e.g. taille moyenne des

arbres, fréquences des taxons, etc.). Une fois que tout cela est fait, on peut (enfin !) utiliser une méthode de super-arbre sur cette forêt ; arrive alors la question majeure suivante : que vaut ce super-arbre ? Est-ce qu'il contient des résolutions aberrantes par rapport à notre connaissance actuelle de la taxonomie ? Est-ce qu'il résout de nouveaux clades ?

Le besoin d'un outil permettant de gérer des collections d'arbres évolutifs s'est donc rapidement fait sentir. Au printemps 2008, j'ai proposé un projet sur ce thème à des étudiants de Master 2 informatique afin de développer un outil pour les besoins de notre équipe. Ce sujet a été choisi par trois étudiants que j'ai encadrés seul : N. Auberval, N. Clairon et S. Diser. Ces étudiants me connaissaient déjà pour avoir suivi un module de Qt (une librairie graphique C++) où ils devaient développer un jeux de casse-briques lors des TP. Leur travail sur ce qui allait devenir PhyloExplorer a largement dépassé mes attentes. Les interactions avec un utilisateur final potentiel, Frédéric Delsuc, ont clairement contribué à maintenir leur motivation. Au fur et à mesure de l'avancé de ce projet, nous avons découvert d'autres utilisations possibles de cet outil qui s'est enrichit de nouvelles fonctionnalités jusqu'au jour de leur soutenance. Devant le potentiel du résultat final, j'ai proposé à Nicolas Clairon de continuer à travailler avec nous, pour un bref CDD de 2 mois, afin de finaliser ce projet et de le passer en phase de production. Parallèlement, Vincent Berry, avait travaillé avec un étudiant (S. Pourali) sur l'utilisation de TREEBASE. Ils avaient notamment fait un effort important pour identifier, à l'aide de TBMap (Page, 2007), les taxons associés aux feuilles des arbres de TREEBASE, qui ont parfois pour label l'identifiant de la séquence utilisée (e.g. ENSRNOG00000003896) et non le nom de l'espèce correspondante (e.g. *Rattus* dans cet exemple). Nous avons donc fusionné nos deux projets en un seul appelé phyloExplorer (`http://www.ncbi.orthomam.univ-montp2.fr/phyloexplorer/`) qui a donné lieu à une publication dans *BMC Evolutionary Biology* et dont les trois étudiants que j'ai encadrés sont co-auteurs.

3.4.2 Quelques fonctionnalités du serveur Web

PhyloExplorer a été développé en séparant la partie "modèle" du projet, qui gère et manipule les données, de la partie graphique qui les affiche. Le respect de ce paradigme important du développement logiciel (connu en tant que "patron de conception" (Gamma et al., 1995) : Modèle, Vue, Contrôleur (Reenskaug, 1979)) fait que PhyloExplorer est à la fois une librairie puissante pour manipuler une collection d'arbres évolutifs et un serveur Web convivial basé sur cette librairie. Cependant je me contenterai, ici, de présenter quelques fonctionnalités phares du serveur Web (`http://www.ncbi.orthomam.univ-montp2.fr/phyloexplorer/`) sans détailler la librairie sous-jacente (`http://code.google.com/p/taxomanie/`).

Les deux formats d'échanges les plus courants pour représenter des arbres phylogénétiques sont le *format newick* et le *format nexus*. La page d'accueil permet de charger une collection d'arbres en utilisant l'un ou l'autre de ces formats. Alternativement, la forêt chargée peut être le résultat d'une requête dans TREEBASE. Le

format de ces requêtes permet d'exprimer simplement des contraintes sur le nombre de représentants d'un groupe taxonomique que doit contenir un arbre pour être sélectionné. Si l'on s'intéresse à la phylogénie des mammifères placentaires (les euthériens) on peut, par exemple, chercher les arbres ayant au moins six taxons de ce groupe et un taxon d'un groupe externe proche (métathérien ou monotrème) pour pouvoir enraciner la phylogénie obtenue. On peut exprimer cette requête de la manière suivante dans phyloExplorer : {eutheria} $>$ 6 and ({metatheria}$>$0 or {monotremata}$>$0).

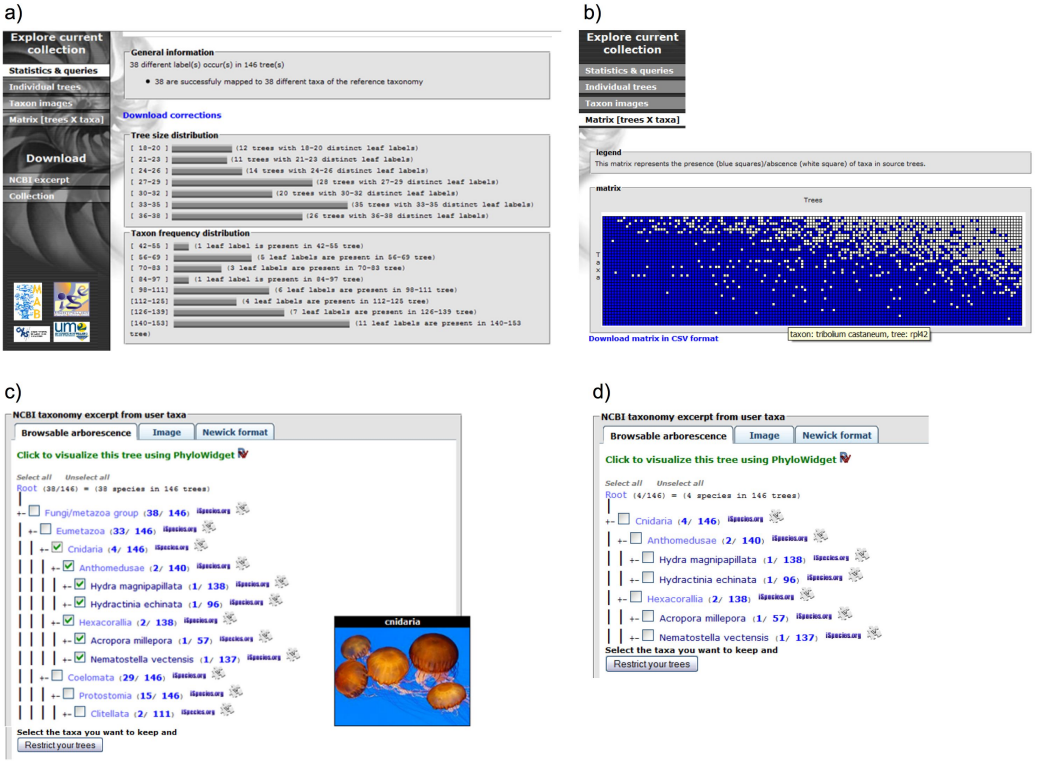

FIGURE 3.13 – **PhyloExplorer : statistiques descriptives d'une collection d'arbres** *Pour une collection donnée, phyloExplorer fournit des statistiques globales a) ainsi qu'une matrice représentant la présence (resp. absence) de chaque taxon dans chaque arbre source b). Une restriction de la taxonomie de référence aux seules espèces présentes dans la collection en cours est également affichée c). Les annotations des nœuds de cette taxonomie permettent, par exemple, de savoir que la collection actuelle contient 4 représentants des Cnidères et que 146 arbres de cette collection contiennent au moins un Cnidère. On peut sélectionner les Cnidères (à l'aide des cases à cocher) et restreindre la collection actuelle à ces seuls taxons ; on obtient alors de nouvelles statistiques d).*

Une fois que l'on a chargé une collection d'arbres, phyloExplorer vérifie la validité des noms taxonomiques associés aux feuilles de ces arbres. Il indique les noms qui ne sont pas reconnus et ceux qui sont ambigus (e.g. *Echinops*, qui est à la fois le nom d'une plante et d'un mammifère) et il aide, au travers de suggestions, à corriger ces problèmes. PhyloExplorer fournit également plusieurs statistiques descriptives per-

mettant notamment de connaître le nombre de représentants de chaque sous-groupe taxonomique présent dans les données (Figure 3.13). Pour cela, phyloExplorer utilise la restriction d'une taxonomie de référence (celle du NCBI [5] ou celle de ITIS [6]) aux seuls taxons présents dans la collection en cours d'analyse. Le simple fait de déplacer la souris près d'un label de cet arbre permet d'obtenir une image du taxon correspondant. Il est également possible, en utilisant des cases à cocher, de sélectionner dans cette taxonomie des espèces d'intérêt et de restreindre les arbres sources à ces seules espèces. On peut aussi utiliser le principe de requêtage décrit plus haut pour sélectionner un sous-ensemble des arbres de la collection. Dans les deux cas, on obtient une nouvelle collection d'arbres et l'ensemble des statistiques est recalculé.

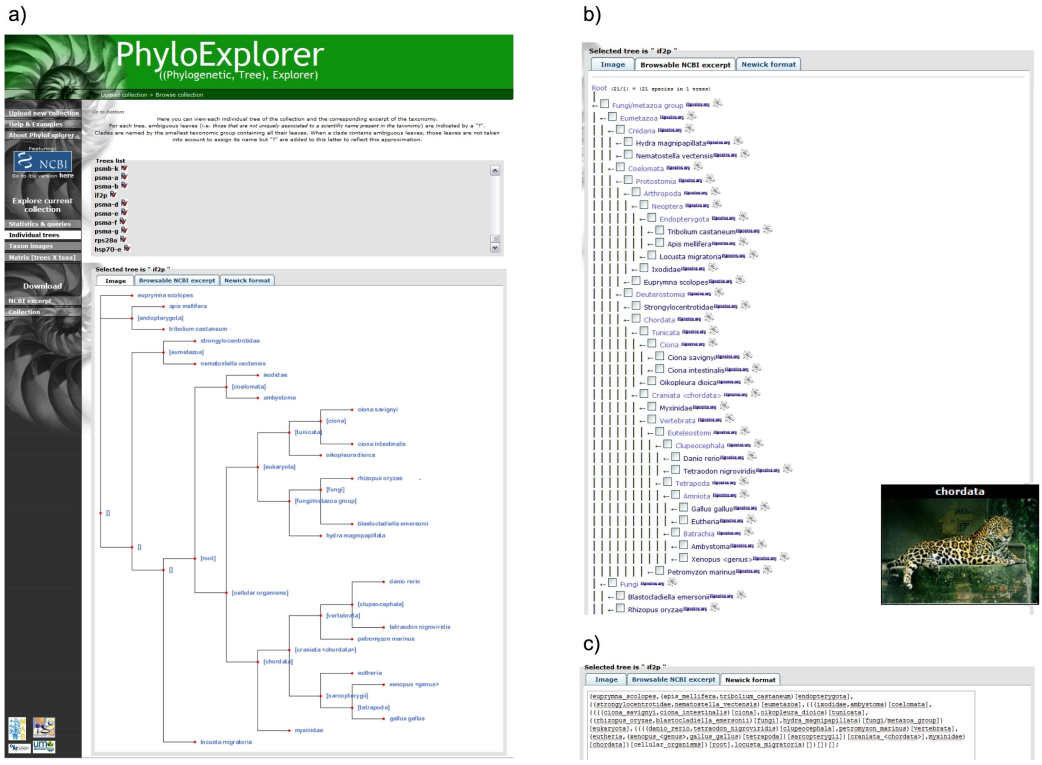

FIGURE 3.14 – **PhyloExplorer : informations disponibles pour chaque arbre source** *PhyloExplorer permet de consulter individuellement les différents arbres de la collection étudiée a) et de voir l'extrait de la taxonomie de référence correspondant b). On peut également obtenir la représentation textuelle (format newick) de cet arbre annoté c).*

PhyloExplorer permet également de consulter les arbres sources individuellement. On dispose alors d'un rendu graphique de l'arbre sélectionné et d'une annotation de ses clades internes. Chaque clade C_i est nommé en utilisant le plus petit groupe taxonomique, de la phylogénie de référence, qui contient l'ensemble des taxons de C_i (Figure 3.14).

5. http://www.ncbi.nlm.nih.gov/Taxonomy/taxonomyhome.html/
6. http://www.itis.gov/

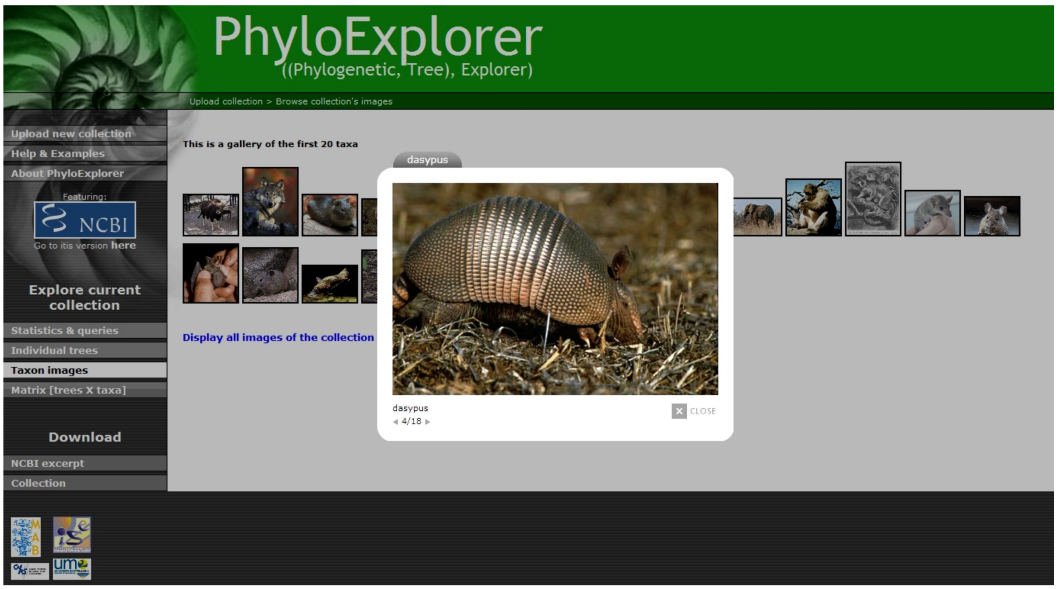

FIGURE 3.15 – **PhyloExplorer : galerie d'images des taxons étudiés**

PhyloExplorer permet aussi de voir des photos (issue de wikispecies [7]) des différents taxons représentés. On peut voir ces images en interagissant avec la représentation hypertexte de la taxonomie (Figure 3.13 c), mais on peut également visualiser et télécharger l'ensemble de ces images sur une page dédiée. Ceci est très pratique pour illustrer une phylogénie (e.g. Figure 2.7) que l'on veut publier ou pour préparer un cours. C'est également fort utile, surtout lorsque comme moi on est un bio-informaticien, pour se faire une idée plus concrète du groupe taxonomique que l'on étudie...

3.5 Conclusion et perspectives

Les méthodes de super-arbre font partie des outils clefs pour assembler l'Arbre de la Vie. Nous avons formalisé deux propriétés qu'elles devraient respecter dans ce contexte. L'étude de ces propriétés de non-contradiction (PC) et d'induction (PI) nous a permis de montrer leurs liens étroits avec la notion d'identification d'un arbre par un ensemble de triplets (Proposition 3.2.8). La preuve que nous avons fournie de ce lien est un argument de plus en faveur de méthodes de super-arbre respectant PC et PI. Le fait d'avoir également prouvé que PC pouvait être garanti sur la seule base des contradictions directes (lemme 3.2.10) nous a permis de développer des méthodes de super-arbre efficaces, respectant PI et PC. De plus, la méthode *PhySIC_IST*, qui résulte de ces travaux, est la seule méthode de véto capable "d'éliminer les conflits" en utilisant à la fois des irrésolutions et le retrait de taxons.

Les méthodes de super-arbre se rangent habituellement dans l'une des deux ca-

7. http://species.wikimedia.org/wiki/Main_Page

tégories suivantes : méthodes de véto ou méthodes de vote. En introduisant un pré-traitement statistiques (STC) qui élimine les résolutions minoritaires des arbres sources, nous avons généré toute une graduation de méthodes de super-arbre (STC + *PhySIC_IST*) qui, selon le paramètre STC utilisé, vont d'une approche essentiellement de type véto à une approche essentiellement de type vote. Les résultats sur des jeux de données simulés et réels confirment la pertinence de STC + *PhySIC_IST*. De plus en découplant, les deux aspects (vote : STC puis véto : *PhySIC_IST*) il nous est possible de fournir un retour à l'utilisateur qui peut l'aider à détecter les gènes (ou les séquences) ayant une évolution atypique (comme lors de notre analyse des *Triticeae*).

Des variantes du pré-traitement STC peuvent également être utiles dans le contexte des réseaux phylogénétiques. Ces réseaux permettent de représenter, en un graphe unique, l'ensemble des clades d'une forêt. En cas de conflits trop nombreux, le réseau peut rapidement devenir incompréhensible. Dans ce cas, un pré-traitement des arbres sources par le STC pourrait, en masquant le signal phylogénétique minoritaire, permettre d'obtenir un réseau plus clair. Céline est maintenant particulièrement à même d'étudier cette application possible du STC puisqu'elle effectue un contrat post-doctoral dans l'équipe de Daniel Huson, qui est internationalement reconnu dans ce domaine. Ils viennent d'ailleurs de co-signer un livre entièrement dédié aux réseaux phylogénétiques (Huson et al., 2010).

Disposant maintenant d'une méthode qui assemble des super-arbres informatifs et fiables, nous avons cherché à l'appliquer sur un jeux de données conséquent afin d'obtenir un super-arbres de plusieurs centaines de taxons couvrant des portions importantes de l'Arbre de la Vie. L'utilisation de phylogénies publiées comme arbres sources s'est révélée décevante. Notamment parce que TREEBASE, le principal projet visant à collecter ces phylogénies, fournit très peu d'informations sur les arbres qu'il contient. Il est donc difficile de savoir les types de méthode (e.g. maximum de parcimonie, maximum de vraisemblance) et de données (e.g. protéiques, nucléiques, morphologiques) utilisés pour inférer chacun de ces arbres. Plus gênant encore, il n'existe pas de distinction entre un arbre présent dans une publication pour illustrer : i) les défauts d'une méthode (ou d'une analyse) précédente ; ii) la spécificité d'un gène ou iii) la phylogénie des espèces proposée par les auteurs. Enfin, la révolution des technologies de séquençage a, là encore, changé la donne. Chaque mois, plusieurs nouveaux génomes complets sont séquencés, si bien que l'on dispose souvent de beaucoup plus d'information sur un groupe taxonomique en analysant directement les données moléculaires qu'en se basant sur des phylogénies publiées.

Pour toutes ces raisons, nous avons donc finalement décidé de privilégier l'assemblage des histoires évolutives de gènes présentes dans des bases de données de séquences homologues. Dans ce cadre, l'utilisation d'OrthoMaM n'a constitué qu'une étape de validation (Scornavacca et al., 2008, figure 8), car elle concerne trop peu d'espèces pour ce type de projet. Par contre, l'utilisation de la base de données HOGENOM (Penel et al., 2009), qui couvre plus de 800 espèces et qui est maintenue par le LBBE (partenaire du projet ANR Phyl-Ariane) offre des perspectives intéressantes.

Cependant, la majorité des familles homologues d'HOGENOM sont multigéniques (i.e. elles contiennent plusieurs représentants d'une même espèce). Les arbres évolutifs de ces familles contiennent donc plusieurs feuilles ayant le même label. Or, aucune méthode actuelle de super-arbres n'est capable d'assembler de tels arbres, ce qui nous a conduit à développer les méthodes décrites dans le prochain chapitre.

Chapitre 4

Prendre en compte les évènements macro-évolutifs

Last night somebody broke into my apartment and replaced everything with exact duplicates... When I pointed it out to my roommate, he said, « Do I know you ? »

Steven Wright

Sommaire

Les arbres évolutifs sont souvent inférés sur la base d'évènements qui se produisent à l'échelle des nucléotides : mutations d'un nucléotide en un autre, insertion ou délétion de quelques nucléotides dans une séquence. Mais de nombreux évènements évolutifs se produisent également à l'échelle des gènes. Pour différencier ces deux types d'évènements nous utiliserons les termes d'*évènements micro-évolutifs* pour faire références aux premiers et d'*évènements macro-évolutifs* pour faire références aux seconds. Les duplications de gènes sont des évènements macro-évolutifs qui ont joué un rôle important dans l'évolution, ayant conduit à la diversité des espèces actuelles (e.g. Ohno, 1970; Lynch and Conery, 2000; Zhang, 2003). Ces évènements de duplication font que de très nombreux gènes sont en plusieurs copies chez certaines espèces. Les arbres évolutifs inférés pour ces gènes contiennent alors plusieurs feuilles étiquetées par une même espèce et un label peut donc apparaître en de multiples endroits de l'arbre. Ces *arbres multi-labels* (ou *arbre* MULT) sont généralement complètement ignorés des analyses phylogénomiques car aucune méthode ne sait les gérer.

Ce chapitre présente nos travaux en cours sur ces arbres MULT. Toutes les solutions algorithmiques décrites dans ce chapitre ont été implémentées à l'aide de la librairie bio++ (Dutheil et al., 2006) que nous avons brièvement présentée dans la section 3.3.1. La première partie de ce chapitre présente différents résultats permettant d'intégrer une partie du signal phylogénétique des arbres MULT dans les analyses phylogénomiques. Dans un premier temps, nous présenterons les concepts "d'auto-isomorphisme" et "d'auto-cohérences" que nous avons introduits dans (Scornavacca et al., 2009) et qui nous permettent d'exploiter le signal phylogénétique de certains arbres MULT (Scornavacca et al., sous presse). Puis nous montrerons sur un exemple l'apport important que l'utilisation de ces arbres MULT peut représenter lors d'une analyse phylogénomique. La seconde partie de ce chapitre décrit l'algorithme que nous avons proposé dans (Doyon et al., 2010) pour inférer le scénario macro-évolutif le plus parcimonieux permettant d'expliquer les différences entre l'histoire évolutive d'un gène et celle des espèces. Il y a un intérêt intrinsèque à inférer les évènements macro-évolutifs subits par un gène, mais ce travail est aussi pour nous une première étape vers une méthode de super-arbre capable d'utiliser directement des arbres MULT. En effet, nous envisageons, à terme, de proposer une méthode de super-arbre dont le critère d'optimisation intégrerait ces coûts de réconciliation.

4.1 Utiliser les familles multigéniques

4.1.1 Motivations

Une *famille de gènes* est constituée d'un ensemble de gènes homologues (i.e. dérivant d'un même gène ancestral). Lorsqu'une espèce possède plusieurs copies de ce gène, on parle de *famille multigénique*. C'est, notamment, le cas de la famille représentée dans la Figure 2.2 qui contient à la fois des gènes qui sont orthologues et paralogues les uns par rapport aux autres. Si l'on utilise les noms d'espèces comme labels des feuilles de l'histoire évolutive de cette famille afin de l'exploiter pour inférer la phylogénie des espèces, on obtient un arbre contenant plusieurs fois le même label (ou *arbre* MULT) – par opposition aux arbres mono-label (ou *arbre* MONO) qui ne contiennent qu'une fois chaque label. Ces arbres MULT, ignorés dans les approches de super-arbres, représentent pourtant 70% des arbres présents dans la version 4 d'HOGENOM. De plus, le problème ne fait que s'accroître avec l'arrivée de nouveaux génomes. En effet, il suffit qu'un seul de ces nouveaux génomes contienne plusieurs copies d'un gène pour que cette famille devienne multigénique et que son histoire évolutive corresponde à un arbre MULT. Ceci explique d'ailleurs que les arbres MULT contiennent, en moyenne, beaucoup plus d'espèces que les arbres MONO. Ces remarques justifient donc, au moins en partie, la critique faite par Dagan and Martin (2006) et reprise par Bapteste et al. (2008) qui qualifiaient les phylogénies obtenues en écartant les familles multigéniques de « Tree of 1% ».

La prise de conscience de ce problème, liée à la publication de Bapteste et al. (2008) et à nos premières analyses d'HOGENOM, nous a amené à étudier ces arbres MULT. En les "regardant" plus en détail, nous avons constaté que dans de nombreux

cas, ces arbres sont constitués de la juxtaposition de deux arbres MONO identiques. Ceci s'explique par un scénario évolutif simple où le gène ancestral a subi une duplication ; puis, ces deux copies ont évolué parallèlement en subissant les mêmes évènements de spéciation. L'arbre multi-labels M de la Figure 4.1 représente schématiquement un tel scénario où le taxon o est un *Outgroup* utilisé pour enraciner l'arbre. Evidemment, il arrive parfois que cette duplication soit plus récente (dans ce cas, seule une partie de l'arbre est dupliquée) ou qu'une copie du gène soit perdue chez une espèce (dans ce cas une copie est incomplète). Ces remarques sont le fondement des notions d'auto-isomorphisme et d'auto-cohérence définies dans cette section. Les travaux sur ces deux notions sont le fruit de la dernière année de thèse de Céline Scornavacca et ont été présentés à la conférence *Language and Automata Theory and Applications 2009* dont les actes sont publiés dans la série *Lecture Notes in Computer Science* (Scornavacca et al., 2009). Suite à cette conférence, notre article a été sélectionné pour une version étendue dans le journal *Information and Computation* (Scornavacca et al., sous presse).

Dans ce chapitre, nous ne considérons que des arbres MULT enracinés et binaires. Ces restrictions peuvent paraître fortes. Cependant, il est généralement possible d'enraciner les arbres phylogénétiques grâce à l'utilisation d'*outgroups* et nous avons d'ailleurs développé un utilitaire nommé bpp-reroot [1] qui facilite l'automatisation de cette tâche. De plus, la plupart des méthodes phylogénétiques classiques (e.g. maximum de vraisemblance) infèrent des arbres binaires. Néanmoins nous sommes conscients que l'enracinement n'est pas toujours trivial et qu'il serait souhaitable de prendre en compte des arbres contenant des irrésolutions (e.g. des arbres obtenus après le retrait de clades ayant une faible valeur de bootsrap, cf section 3.3.2.3). Nous travaillons actuellement sur ces deux aspects.

4.1.2 Extraire le signal de spéciation des arbres MULT

Les résultats de cette section reposent sur la notion de nœuds de duplication observable. Après avoir défini cette notion nous présenterons les solutions algorithmiques qui nous permettent d'exploiter une partie du signal de spéciation d'arbres MULT qui peuvent être auto-isomorphes, auto-cohérents ou quelconques.

4.1.2.1 Caractériser et identifier les nœuds de duplication observée

Les arbres MULT sont une généralisation des arbres MONO. En particulier, dans le cas d'un arbre MULT M, il est important de distinguer l'ensemble $\mathcal{L}(M)$ de ses feuilles, l'ensemble $L(M)$ des labels de ses feuilles (e.g. le nom des espèces étudiées) et le "multi-ensemble" $L_m(M)$ de ces mêmes labels. Un *multi-ensemble* (*multiset* en anglais) est une généralisation de la notion d'ensemble qui autorise la présence de plusieurs éléments identiques. A titre d'exemple, les deux arbres de la Figure 4.1 ont le même ensemble de labels : $L(M) = L(M_2) = \{a, b, c, d, o\}$ mais pas les mêmes multi-ensembles puisque $L_m(M) = \{a, b, c, d, c, b, a, o\}$ tandis que $L_m(M_2) =$

1. http://home.gna.org/bppsuite/

$\{a, d, c, b, o\}$ (N.B. dans un multi-ensemble, tout comme dans un ensemble, l'ordre d'écriture des éléments n'a pas d'importance).

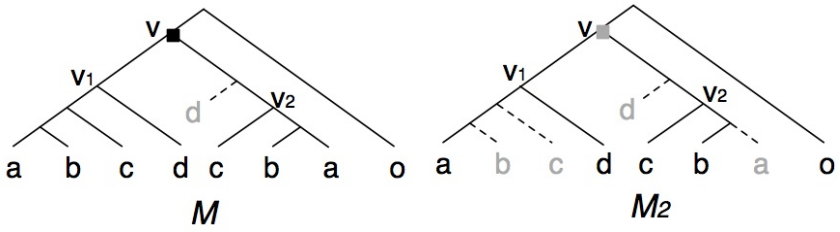

FIGURE 4.1 – **Exemple d'arbres évolutifs implicant des duplications.** *Les arbres de gènes observés sont représentés par des traits pleins noirs. L'histoire évolutive d'une famille multigénique peut inclure des pertes (représentées en pointillé et en gris) qui ne sont pas visibles dans les arbres de gènes inférés et des duplications (représentées par des carrés). Dans l'arbre* MULT *M, la duplication qui s'est produite au nœud v est révélée par la présence des mêmes labels dans ses sous-arbres gauche (M_{v1}) et droit (M_{v2}). Cette duplication reste détectable malgré la perte d'une copie chez l'espèce d ; mais des pertes asymétriques et nombreuses, comme celles de M_2, peuvent complètement masquer un évènement de duplication.*

Etant donné un nœud v d'un arbre MULT M, nous utiliserons les notations suivantes tout au long de ce chapitre : si v est une feuille l_v désignera son label ; si v est un nœud interne v_1 et v_2 désigneront ses deux fils et $fils(v)$ désignera l'ensemble $\{v_1, v_2\}$; enfin, M_v désignera le sous-arbre ayant v pour racine (e.g., dans la Figure 4.1, $M_{v2} = ab|c$). Pour alléger les notations, nous utiliserons $L_m(v)$ et $L(v)$ pour désigner respectivement $L_m(M_v)$ et $L(M_v)$ lorsqu'il n'y a pas d'ambiguïté sur M. Ces notations nous permettent de définir formellement ce qu'est pour nous un "nœud de duplication observée".

Définition 4.1.1 – nœud de duplication observée (odn) : *Un nœud v de M est un nœud de duplication observée (ou* ODN*) si, et seulement si, l'intersection de $L_m(v_1)$ et $L_m(v_2)$ n'est pas vide.*

Il découle directement de cette définition qu'un ODN est un nœud pour lequel le même label est utilisé à la fois dans son sous-arbre droit et dans son sous-arbre gauche. Ceci est caractéristique d'un évènement de duplication (e.g. le nœud v de l'arbre M_2 de Figure 4.1). Mais il est possible qu'il existe d'autres nœuds de l'arbre où des duplications se sont produites mais ne sont pas "observées" à cause de pertes de gènes ultérieures (e.g. le nœud v de l'arbre M_2 de Figure 4.1). Pour toutes les analyses présentées dans ce chapitre et qui s'appuient sur ces ODN, il est préférable de privilégier des génomes complètement séquencés. En effet, pour les autres génomes il est possible qu'une seule des multiples copies d'un gène ait été séquencée ce qui rend inobservables certains nœuds de duplication.

Un algorithme simple permet de calculer l'ensemble $\mathcal{D}(M)$ des nœuds de duplication d'un arbre M en $O(n^2)$ où $n = |\mathcal{L}(M)| = |L_m(M)|$. En effet, en suivant la

définition, il suffit pour chacun des $O(n)$ nœuds internes de M de calculer l'intersection de deux ensembles de $O(n)$ éléments pour décider si ce nœud est, ou non, dans $\mathcal{D}(M)$. Nous avons proposé une solution linéaire à ce problème qui s'appuie sur le calcul du plus petit ancêtre commun de deux nœuds x et y noté $lca(x, y)$ (pour *least common ancestor*). La preuve du lemme 4.1.2 sur lequel s'appuie notre algorithme, est disponible dans (Scornavacca, 2009; Scornavacca et al., sous presse) – tout comme celles des autres résultats théoriques de cette section.

Lemme 4.1.2 *Un nœud v est un* ODN *d'un arbre M si, et seulement si, il est le lca d'au moins un couple de feuilles m et p de M ayant le même label (i.e. $v = lca(m, p)$ et $l_m = l_p$).*

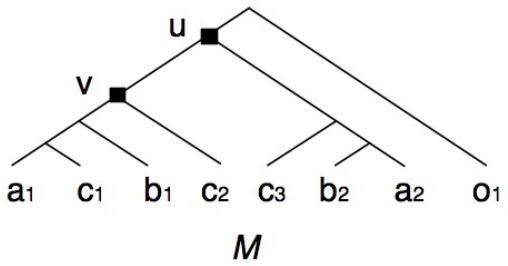

FIGURE 4.2 – **Principe de l'algorithme linéaire permettant d'identifier les ODN.** *Dans cet exemple les feuilles de l'arbre M, dont le multi-ensemble des labels est initialement $L_m(M) = \{a, c, b, c, c, b, a, o\}$, sont ré-étiquetées. Après cette opération, la nouvelle étiquette d'une feuille est constituée de son label initial suivi de son numéro d'occurrence. Les deux* ODN *u et v sont des lca d'occurrences consécutives de feuilles ayant le même label initial : $v = lca(c_1, c_2)$ et $u = lca(c_2, c_3)$.*

Ce lemme n'est pas suffisant pour identifier les ODN en $O(n)$, car même en utilisant le pré-traitement linéaire proposé par Harel and Tarjan (1984) qui permet ensuite de connaître le *lca* de deux nœuds en $O(1)$, il reste, *a priori*, nécessaire de calculer le *lca* des $O(n^2)$ paires de feuilles. Cependant, il n'existe que n ODN possibles et de nombreuses paires de feuilles partagent le même lca. Nous avons montré qu'il suffit de tester un nombre linéaire de paires de feuilles pour identifier l'ensemble des ODN. Pour savoir quelles sont ces paires, il faut ré-étiqueter les feuilles en utilisant leur label initial suivi de leur "numéro d'occurrence" en considérant les feuilles de gauche à droite[2]. La Figure 4.2 fournit un exemple de ré-étiquetage des feuilles suivant ce principe. La solution linéaire que nous avons proposée, détaillée dans l'algorithme 4, repose sur le lemme 4.1.3, qui garantit que tout ODN est le lca de deux occurrences consécutives du même label. Par exemple, l'arbre M de la Figure 4.2 a deux ODN u et v tels que $u = lca(c_1, c_2)$ et $v = lca(c_2, c_3)$.

Lemme 4.1.3 *Soit M un arbre* MULT. *Le nœud v est un* ODN *de M (i.e. $v \in \mathcal{D}(M)$) si, et seulement si, il existe deux occurrences consécutives d'un label e, notées e_i et e_{i+1}, telles que $v = lca(e_i, e_{i+1})$.*

2. ceci se fait simplement par un parcours d'arbre en profondeur.

Algorithm 4: Calcul linéaire des ODN d'un arbre M

Data: Un arbre MULT M.
Result: L'ensemble $\mathcal{D}(M)$ des ODN de M.
Ré-étiquetter les feuilles de M en utilisant leur label suivi de leur numéro d'occurrence (en considérant les feuilles de gauche à droite);
Effectuer le prétraitement linéaire de Harel & Tarjan (1984);
$\mathcal{D}(M) \leftarrow \emptyset$;
foreach *"label répété" e* **do**
  foreach $(\{e_j, e_{j+1}\})$ **do** $\mathcal{D}(M) \leftarrow \mathcal{D}(M) \cup lca(e_j, e_{j+1})$;

L'identification de ces nœuds de duplication apparente est à la base des notions d'auto-isomorphise et d'auto-cohérence qui sont présentées dans les sections suivantes.

4.1.2.2 Caractériser et vérifier l'auto-isomorphisme d'un arbre MULT

La présence d'un ODN v conduit à observer deux sous-arbres M_{v1} et M_{v2} qui représentent l'évolution de deux copies du même gène. Ces deux histoires peuvent différer du fait de résolutions différentes ou de pertes asymétriques de gènes (e.g. la perte de d dans l'arbre M de la Figure 4.1). Cependant, ces deux histoires sont issues des mêmes évènements de spéciation et sont donc souvent très semblables. Si ces deux copies sont "isomorphes" (i.e. topologiquement identiques) on peut ne conserver qu'une d'elles sans perdre d'information sur la phylogénie des espèces de M. Ce faisant, on obtient un arbre M_1 ayant un ODN de moins que l'arbre initial M. On peut alors étudier un ODN de M_1 et, si ses sous-arbres sont isomorphes, produire un nouvel arbre M_2 ayant encore un ODN de moins. S'il est possible, par une série de simplifications de ce type, d'obtenir un arbre MONO T à partir d'un arbre MULT M, on dit que M est *auto-isomorphe* (Figure 4.3) et que T est une *simplification isomorphe* de M (il est facile de vérifier que toutes les simplifications isomorphes d'un arbre MULT M sont isomorphes entre elles). L'avantage est que T représente fidèlement le signal de spéciation contenu dans M et qu'il peut être utilisé comme arbre source d'une méthode de super-arbre. L'obtention de l'arbre T s'appuie sur le test d'isomorphisme de deux arbres MULT pour lequel nous avons proposé la solution linéaire décrite ci-dessous.

Définition 4.1.4 – **isomorphisme d'arbres MULT** : *Deux arbres* MULT, M_1 *et* M_2, *sont isomorphes si, et seulement si, il existe une bijection entre les nœuds de* M_1 *et ceux de* M_2 *qui préserve les labels des feuilles et les parentés entre nœuds.*

Sur la base du lemme 4.1.5, Berry and Nicolas (2006) ont décrit un algorithme permettant d'établir l'isomorphisme de deux arbres MONO en considérant leurs *cerises*, i.e. leurs nœuds internes v ayant deux feuilles pour "fils" ($|L_m(T_v)| = 2$). Le nom de cerise vient de la forme en ${}_o\wedge_o^v$ similaires aux paires de cerises qui servent parfois de boucles d'oreilles aux (grands) enfants.

Lemme 4.1.5 (Gusfield (1991)) *Si deux arbres* MONO T_1 *et* T_2 *sont isomorphes et que* c_1 *est une cerise de* T_1, *alors il existe une cerise* c_2 *de* T_2 *telle que* $L(c_1) = L(c_2)$

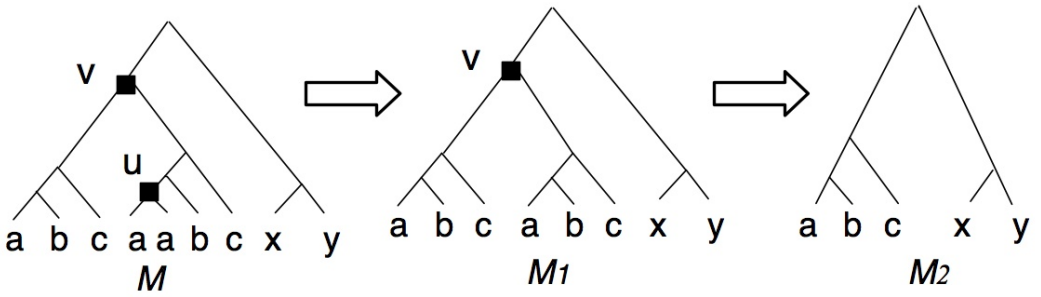

FIGURE 4.3 – **Notion d'auto-isomorphisme d'un arbre MULT.** *L'arbre* MULT *M contient deux* ODN *u et v. Les deux sous-arbres M_{u1} et M_{u2} sont isomorphes ; en supprimant un de ces sous-arbres on obtient l'arbre* MULT *$M1$ qui ne contient plus qu'un seul odn. On peut encore simplifier $M1$ en ne gardant qu'un des deux sous-arbres isomorphes $M1_{v1}$, $M1_{v2}$. Cette dernière simplification conduit à l'arbre $M2$ qui est* MONO *; M est donc auto-isomorphe.*

Les choses se compliquent un peu dans le cas d'arbre MULT, car une même cerise peut apparaître plusieurs fois (e.g. ${}_a\wedge_b$ apparaît deux fois dans l'arbre $M1$ de la Figure 4.3). Dans le cas d'un arbre MULT M, nous parlerons donc de multi-cerise pour désigner la liste des cerises ayant les deux mêmes labels, et l'on notera $|mc|$ son nombre d'occurrence dans M. Cette terminologie nous permet de généraliser le lemme 4.1.5 au cas des arbres MULT et d'obtenir une solution en $O(n)$ pour vérifier l'isomorphisme de deux arbres MULT (Algorithme 5).

Lemme 4.1.6 *Si deux arbres* MULT *M_1 et M_2 sont isomorphes et que mc_1 est une multi-cerise de T_1, alors il existe une multi-cerise mc_2 de M_2 telle que $L(mc_1) = L(mc_2)$ et $|mc_1| = |mc_2|$.*

Les preuves de complexité et de correction de l'algorithme 5 sont détaillées dans (Scornavacca et al., sous presse). L'idée est de "grignoter" les cerises des arbres M_1 et M_2 dont on veut vérifier l'isomorphisme. Lorsqu'une cerise mc_1 de M_1 correspond à une cerise mc_2 de $M2$, et que $|mc_1| = |mc_2|$, toutes les occurrences de mc_1 et mc_2 sont remplacées par une feuille ayant un nouveau label l_{new}. Ceci engendre de nouvelles multi-cerises, qui peuvent à leur tour être grignotées. Le "grignotage" s'arrête lorsqu'il ne reste plus de multi-cerises (M_1 et M_2 sont isomorphes) ou lorsqu'on n'arrive pas à "grignoter" une des multi-cerises (M_1 et M_2 ne sont pas isomorphes).

Algorithm 5: Vérification linéaire de l'isomorphisme de deux arbres MULT

Data: Deux arbres MULT M_1 et M_2.
Result: Vrai si M_1 et M_2 sont isomorphes, faux sinon.
$L_{mc} \leftarrow$ les multi-cerises de M_1 et de M_2;
Construire un dictionnaire H (avec une structure de "hachage") dont les clefs sont les éléments $mc \in L_{mc}$;
A chaque clef mc de H associer les listes L^1_{mc} et L^2_{mc}, contenant respectivement les occurrences de mc dans M_1 et M_2;
while $(L_{mc} \neq \emptyset)$ **do**
    $mc \leftarrow$ removeFirst(L_{mc});
    if $(|L^1_{mc}| = |L^2_{mc}|)$ **then**
        Créer un nouveau label l_{new};
        Remplacer toutes les cerises de L^1_{mc} et L^2_{mc} par des feuilles ayant le label l_{new};
        Ajouter les nouvelles multi-cerises à la fin de L_{mc};
        Mettre à jour H;
    else return FAUX;
return VRAI;

L'Algorithme 5 est au cœur de la procédure récursive qui nous permet de vérifier en un temps linéaire qu'un arbre MULT est auto-isomorphe. Il suffit pour chaque nœud de duplication $v \in \mathcal{D}(M)$ de s'assurer que les deux sous-arbres M_{v1} et M_{v2} sont isomorphes. Une légère adaptation de cette procédure permet (en ne conservant qu'une des deux copies isomorphes) de construire, également en un temps linéaire, une simplification isomorphe maximale[3] de M (Scornavacca et al., sous presse, Algorithme3). Quand M est auto-isomorphique, cette simplification maximale correspond à un arbre MONO utilisable par les méthodes de super-arbres.

4.1.2.3 Caractériser et vérifier l'auto-cohérence d'un arbre MUL

Lorsqu'un arbre MULT M n'est pas auto-isomorphe, on peut se demander si cela vient du fait que les deux histoires obtenues à la suite d'une duplication sont partielles (mais cohérentes entre elles), ou si elles contiennent des informations contradictoires. Dans le premier cas on dira que M est auto-cohérent dans le second cas il ne l'est pas. La Figure 4.4 montre un exemple de chacun de ces cas, l'arbre M_1 y est auto-cohérent tandis que l'arbre M_2 ne l'est pas. L'idée de l'auto-cohérence est de s'assurer que le signal de spéciation de l'arbre MULT est cohérent et que l'on n'a pas d'informations contradictoires selon les copies du gène que l'on considère. Pour cela, on peut se baser sur les triplets et vérifier que l'ensemble des triplets de M est compatible. Evidemment dans un arbre MULT, si on a une duplication à un nœud v qui conduit à observer les taxons $\{a, b, c\}$ dans M_{v1} et dans M_{v2} ; l'arbre M représente tous les triplets possibles sur ces trois taxons. Par exemple, pour obtenir $bc|a$ à partir de l'arbre M de la Figure 4.4, il suffit de prendre les instance de b et de c présentes dans M_{v1} et de prendre celle de a dans M_{v2}. Cependant, se faisant, on ne considère pas des instances orthologues (puisque leur histoire implique le nœud de duplication v). Si l'on suppose que les ODN sont effectivement des évènements de duplication, et que ce sont les seuls, alors on peut différencier les triplets d'instances

3. au sens où il n'existe pas de simplification isomorphe de M ayant moins de feuilles

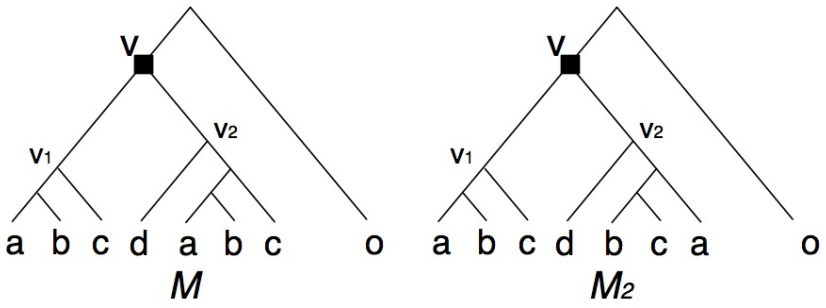

FIGURE 4.4 – **Notion d'auto-cohérence d'un arbre MULT.** *L'arbre* MULT *M n'est pas auto-isomorphe (car d n'est pas présent dans M_{v1}, mais il est auto-cohérent car le signal de spéciation des deux copies issu de v est cohérent. L'arbre $M2$ n'est ni auto-isomorphe (pour la même raison que M) ni auto-cohérent. En effet, dans $M2$ l'histoire de la première copie (i.e. $M2_{v1}$) contient le triplet $ab|c$ tandis que l'histoire de la seconde copie (i.e. $M2_{v2}$) contient le triplet $bc|a$.*

orthologues de triplets d'instances paralogues. Les premiers sont évidemment ceux qui nous intéressent pour inférer l'histoire des espèces et ce sont eux qui doivent être compatibles pour que M soit dit auto-cohérent. Ces triplets d'orthologues sont ceux qui n'impliquent pas d'ODN, d'où l'appellation de $\mathcal{R}_{wd}(M)$ pour désigner l'ensemble les contenant (wd : *Without Duplication*).

Définition 4.1.7 – $R_{wd}(M)$ **:** *Soit M un arbre* MULT*, l'ensemble $R_{wd}(M)$ des triplets sans duplication est constitué des triplets $ab|c$ pour lesquels il existe trois feuilles $x, y, z \in \mathcal{L}(M)$ avec $l_x = a$, $l_y = b$, $l_z = c$ et telles que*

(i) $lca(x, y) \neq (lca(x, z) = lca(y, z))$

(ii) $lca(\{x, y, z\}) \notin \mathcal{D}(M)$ *et* $lca(x, y) \notin \mathcal{D}(M)$

Le point (i) assure que $ab|c$ est bien représenté par M, tandis que le point (ii) assure qu'aucun des deux nœuds internes de ce triplet n'est un nœud de duplication. A titre d'exemple, les ensembles R_{wd} des arbres M et $M2$ de la Figure 4.4 sont :

- $R_{wd}(M) = \{ab|c,\ ab|d,\ bc|d,\ ab|o,\ ac|o,\ ad|o,\ bc|o,\ bd|o,\ cd|o\}$
- $R_{wd}(M2) = \{ab|c,\ ab|d,\ bc|d,\ ab|o,\ ac|o,\ ad|o,\ bc|o,\ bd|o,\ cd|o,\ bc|a\}$

Définition 4.1.8 – **auto-cohérence :** *Un arbre* MULT *est dit auto-cohérent si, et seulement si, l'ensemble $\mathcal{R}_{wd}(M)$ est compatible, i.e. il existe un arbre* MONO *T tel que $\mathcal{R}_{wd}(M) \subseteq tr(T)$.*

L'étude du graphe de Aho (cf section 3.3.1) d'un ensemble de triplets R permet de savoir en $O(|R|)$ si cet ensemble est compatible. Comme l'ensemble $R_{wd}(M)$ contient $O(n^3)$ triplets, cette approche donne une solution simple pour vérifier l'auto-cohérence d'un arbre MULT en $O(n^3)$. Cependant, nous avons prouvé que cette vérification pouvait se faire en $O(n.log^2(n))$ en utilisant un sous-ensemble $R^l_{wd}(M) \subseteq R_{wd}(M)$ de taille linéaire [4] en fonction du nombre d'espèces de M. Ceci

4. d'où le l de la notation

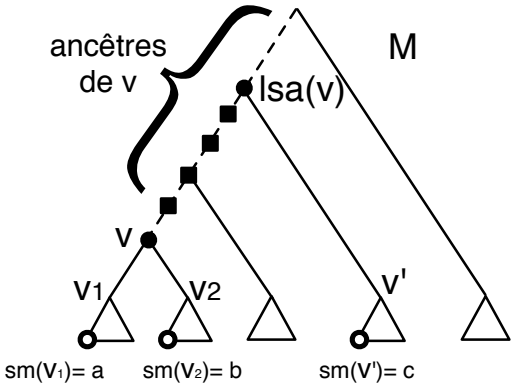

FIGURE 4.5 – **Principe de la représentation linéaire de $\mathcal{R}_{wd}(M)$.** *Le choix du triplet $ab|c$ comme représentant du nœud de spéciation v s'appuie sur l'identification du premier ancêtre de v qui n'est pas un* ODN *(les odns sont repésentés par des ■) et sur une fonction d'ordre sur les feuilles de M (voir le corps du texte pour plus de détails).*

est lié aux règles d'induction évoquées au chapitre précédent (3.2.1) qui font qu'un ensemble contenant peu de triplets peut néanmoins en induire un grand nombre. D'une certaine manière notre résultat sur $R^l_{wd}(M)$ est une généralisation du fait que tout arbre (MONO) binaire T peut être représenté de manière équivalente par un nombre linéaire de triplets (Steel, 1992). La représentation linéaire que nous utilisons nécessite d'introduire un ordre total sur les feuilles de M. On peut alors définir, pour chaque nœud v, $sm(v)$ le label de la plus petite (*smallest*) feuille appartenant à M_v (Figure 4.5). Comme les triplets de $R_{wd}(M)$ sont liés à des nœuds de spéciation (i.e. des nœuds qui ne sont pas des ODN), le premier parent de v qui soit un nœud de spéciation joue un rôle particulier. On note ce nœud $lsa(v)$ (pour *least speciation ancestor*) et l'on notera v' le fils de $lsa(v)$ tel que $v \notin M_{v'}$. On obtient l'ensemble $R^l_{wd}(M)$ en prenant pour chaque nœud de spéciation v le triplet $sm(v_1)sm(v_2)|sm(v')$. A titre d'exemple, si l'on prend comme ordre des feuilles l'ordre de lecture (i.e. de gauche à droite), les ensembles R^l_{wd} des arbres M et M_2 de la Figure 4.4 sont :

- $R^l_{wd}(M) = \{ab|c,\ ac|o,\ ac|d,\ cd|o\}$
- $R^l_{wd}(M2) = \{ab|c,\ ac|o,\ bc|a,\ ba|d,\ ad|o\}$

Quand un arbre MULT n'est pas auto-cohérent cela indique que son signal de spéciation n'est pas totalement fiable. Dans un premier temps, nous avons donc préféré ignorer ces arbres dans nos analyses. Par contre, si un arbre M est auto-cohérent, il nous semble souhaitable d'intégrer son signal de spéciation (i.e. les triplets de R_{wd}) dans nos forêts sources en utilisant un arbre MONO T. Pour cela, il faut que T représente fidèlement le signal de M au sens où tous ses triplets sont induits par le signal de M et aucun d'eux ne le contredit. On cherche donc à construire un arbre T satisfaisant PC et PI pour R_{wd} ce que l'on peut faire en utilisant *PhySIC* sur l'ensemble R^l_{wd}. On peut également envisager d'utiliser *PhySIC_IST* mais, tandis

que *PhySIC* n'utilise que les triplets, *PhySIC_IST* exploite la structure des arbres sources pour calculer efficacement les supports de branches lors de l'insertion d'un taxon (cf section 3.3.2.2).

Une manière alternative de produire un arbre MONO à partir d'un arbre MULT M est de choisir d'ignorer une copie de chaque duplication. On obtient alors un arbre T MONO ne contenant que des feuilles orthologues. Suivant les choix qui sont faits, l'arbre T obtenu peut être plus ou moins informatif.

4.1.2.4 Identifier le sous-arbre d'orthologues le plus informatif d'un arbre MULT

Qu'un arbre MULT M soit ou non auto-cohérent, l'identification des ODN permet d'identifier les triplets d'orthologues de M. Si pour chaque ODN $v \in \mathcal{D}(M)$ on choisit de garder un seul des deux sous-arbres M_{v1}, M_{v2}, on obtient un arbre MONO qui ne contient que des orthologues (apparents [5]). On appelle cette procédure, *l'élagage (des duplications)*.

Définition 4.1.9 – élagage (des duplications) : *Soit M un arbre* MULT *et T un arbre* MONO. *On dit que T est obtenu par élagage (des duplications) de M (noté $T \lesssim M$) si, et seulement si, T peut être obtenu à partir de M en choisissant pour chacun de ses* ODN *v soit M_{v_1} soit M_{v_2} et en éliminant de M tous les sous-arbres non choisis.*

Pour un arbre MULT binaire M, on peut obtenir en un temps linéaire un arbre MONO T_{max} qui est un élagage de M d'informativité maximale (au sens du CIC présenté dans la section 3.3.2.1). Il suffit pour cela de garder à chaque fois le sous-arbre contenant le maximum de labels distincts. Cet arbre T_{max} peut être utile dans un contexte de super-arbre mais également dans le cadre d'une analyse de type supermatrice. En effet, si l'on dispose des séquences moléculaires utilisées pour inférer l'arbre M, comme c'est le cas pour HOGENOM, les feuilles de T_{max} correspondent aux plus grands sous-ensembles de séquences orthologues disponibles.

On peut également généraliser ce problème au cas d'une forêt $\mathcal{M}$ d'arbres MULT pour laquelle on cherche à élaguer les arbres de $\mathcal{M}$ de manière à obtenir une forêt d'arbres MONO compatible. Dans ce cas, chaque choix fait pour un ODN v d'un arbre M se fait en prenant en compte la cohérence des deux histoires M_{v1} et M_{v2} avec celles qui sont présentes dans les autres arbres MULT de $\mathcal{M}$. Ce qui correspond à résoudre le problème suivant :

Problème	Existence of a Pruned and Compatible Forest (EPCF)
Données	Une forêt d'arbres MULT $\mathcal{M} = \{M_1, \cdots M_k\}$.
Résultat	Une forêt d'arbres MONO $\mathcal{F} = \{T_1, \cdots T_k\}$ telle que : (i) pour tout i entre 1 et k : $T_i \lesssim M_i$. (ii) $\mathcal{F}$ est compatible.

5. comme nous l'avons expliqué sur l'exemple de la Figure 4.1 on peut ne pas observer tous les nœuds de duplication, et dans ce cas l'erreur se répercute sur la prédiction d'orthologie

Nous avons montré (par une "réduction" de EPCF au problème classique "3-SAT") que ce problème, est NP-complet.

4.1.3 Application à HOGENOM

Les résultats et algorithmes de la section précédente permettent d'extraire sous forme d'arbres MONO une partie du signal phylogénétique d'une forêt $\mathcal{M}$ d'arbres MULT. Nous avons conduit une série d'expériences afin d'évaluer le gain qui résulte de l'ajout de ces arbres MONO lors de l'inférence, via une approche de type super-arbre, de la phylogénie des espèces qui sont présentes dans $\mathcal{M}$. Pour cela, nous avons comparé plusieurs forêts $\mathcal{F}$ d'arbres MONO qui peuvent être dérivées de $\mathcal{M}$:

- F_1 : la forêt contenant les arbres MONO de $\mathcal{M}$;
- F_2 : la forêt contenant les arbres MONO obtenus par simplifications isomorphiques des arbres de $\mathcal{M}$ qui sont auto-isomorphes mais pas MONO ;
- F_3 : la forêt contenant les arbres MONO qui résument le signal d'arbres de $\mathcal{M}$ qui sont auto-cohérents mais pas auto-isomorphes. Pour ces arbres, nous avons testé les deux solutions évoquées plus haut :
  - F_3^p : contient les arbres obtenus par les élagages (*Prunning*) isomorphiques maximisant l'informativité des arbres MONO obtenus ;
  - F_3^s : contient les arbres correspondant aux résumés (*Summaries*), produits par *PhySIC*, des ensembles *Rwd*.
- F_{all} : la forêt obtenue en prenant l'union des précédentes. Il en existe donc deux variantes :
  - $F_{all}^p = F_1 \cup F_2 \cup F_3^p$
  - $F_{all}^s = F_1 \cup F_2 \cup F_3^s$

4.1.3.1 Un super-arbre du vivant

Dans un premier temps, nous avons considéré la totalité des arbres contenus dans la version 4 de la base de données HOGENOM (Penel et al., 2009). Cette version d'HOGENOM couvre 381 espèces (correspondant à des génomes complètement séquencés) et contient 147 586 familles de gènes pour lesquelles on dispose des alignements de séquences et des arbres inférés par maximum de vraisemblance à partir de ces alignements. Parmi ces arbres, nous n'avons considéré que les 46 535 qui contiennent au moins trois espèces (les autres n'apportent pas d'information sur la phylogénie des espèces). Nous avons également dû écarter de notre étude, 116 de ces arbres qui n'étaient pas binaires. Notre forêt initiale $\mathcal{M}$ contenait donc 46 419 arbres (dont 33 041 arbres MULT) et couvrait 376 espèces. A partir de cette forêt, nous avons construit les forêts F_1, F_2, F_3^s, F_3^p, F_{all}^s et F_{all}^p. Grâce aux faibles complexités de nos algorithmes, cette étape ne prend que quelques minutes sur une machine linux standard [6] en utilisant les implémentations de nos algorithmes disponibles à l'adresse : `http://www.atgc-montpellier.fr/ssimul/`. Par exemple, la détection d'ODN pour l'ensemble des 46 335 arbres de $\mathcal{M}$ ne prend que ∼2min,

6. processeur 3GHz et 4Go de RAM

l'élagage isomorphique des 17 674 de F_3 se fait en ∼10s et le traitement de cette même forêt par *PhySIC* nécessite ∼20min. Dans le cas de l'élagage, si l'arbre obtenu contient moins de trois feuilles, il n'est pas gardé, ceci explique qu'il y ait quelques arbres de moins dans F_3^p que dans F_3^s.

TABLE 4.1 – **Quelques statistiques sur différentes forêts d'arbres MONO dérivées des arbres MULT d'HOGENOM.**

	F_1	F_2	F_3^s	F_3^p	F_{all}^s	F_{all}^p
nb arbres	13,378	11,891	17,674	16,148	42,943	41,417
nb total de triplets	151,287	2×10^6	421×10^6	424×10^6	423×10^6	426×10^6
nb de triplets distincts	68,538	601,429	22.9×10^6	22.2×10^6	22.9×10^6	22.3×10^6
% de triplets représentés	0.3%	2.3%	86.8%	84.4%	86.9%	84.4%

Afin d'évaluer le gain d'information apporté par l'utilisation des arbres MULT, nous avons calculé quelques statistiques permettant d'estimer le signal phylogénétique des différentes forêts MONO dérivées de ces arbres. Ces statistiques sont résumées dans le tableau 4.1. Ces résultats montrent que les arbres MONO de M (F_1) représentent toujours une part importante des arbres que nous sommes capables d'intégrer dans notre analyse (∼13 000 arbres dans F_1 et ∼42 000 dans F_{all}). Par contre, les arbres de F_1 sont nettement moins informatifs que ceux de F_2 ou F_3 ; e.g. le nombre total de triplets ($|tr(F)|$) est de ∼150 000 pour F_1 alors qu'il est de l'ordre de 2 millions pour F_2 et de l'ordre de 400 millions pour F_3. Ceci est encore plus frappant lorsque l'on considère le pourcentage des groupes de trois espèces pour lesquels il existe au moins un triplet dans F. Ce pourcentage est de $0,3\%$ pour F_1 et de $2,3\%$ pour F_2 alors qu'il est d'environ 85% pour les forêts F_3^s et F_3^p qui intègrent les arbres auto-cohérents. Il semble donc bien que la prise en compte de ces arbres MULT apporte un supplément d'information capital pour l'inférence de l'arbre des espèces.

Nous avons ensuite utilisé MRP et *PhySIC_IST* sur chacune de ces forêts et comparé l'informativité (le CIC) des super-arbres inférés. Les résultats obtenus confirment l'intérêt d'exploiter les arbres MULT de la forêt initiale. En effet, lorsque l'on utilise STC+*PhySIC_IST*, on passe d'un super-arbre ayant un CIC de 3% pour F_1 à un super-arbre ayant un CIC de 51% pour F_{all}^p et un CIC de 57% pour F_{all}^s. Le gain est aussi remarquable avec MRP. La forêt F_1 est si peu informative que nous n'avons pas pu obtenir de super-arbre avec MRP à partir de cette forêt (nous avons interrompu le programme au bout d'une semaine de calcul). En revanche MRP fournit (rapidement) des super-arbres très informatifs (CIC ∼100%) en utilisant l'une ou l'autre des forêts F_{all}. Il semble également que la qualité de l'arbre s'améliore avec l'augmentation de l'information. De fait, même si les CIC sont déjà très bons avec les forêts F_3, certains taxons sont mieux placés dans les super-arbres obtenus avec les forêts F_{all}. Ces résultats sont commentés de manière plus détaillée dans Scornavacca et al. (sous presse). Afin de voir de manière plus concrète l'apport des arbres MULT dans une analyse de type super-arbre, nous avons conduit une analyse

similaire sur un groupe taxonomique beaucoup plus restreint et pour lequel il est possible d'inclure les super-arbres obtenus dans ce manuscrit.

4.1.3.2 Un super arbre des eucaryotes

La validation d'un arbre du vivant de près de 400 espèces est assez complexe. Nous avons donc conduit une analyse similaire à celle décrite ci-dessus mais en nous limitant aux eucaryotes. Cela élimine une grande partie des espèces d'HOGENOM qui sont des bactéries ou des archées. Mais cette analyse inclut tout de même les animaux, les plantes, les champignons et les protistes (un groupe d'eucaryotes unicellulaires très divers) présents dans HOGENOM, soit un total de 45 espèces. Pour cette analyse, nous avons utilisé PhyloExplorer afin de ne conserver que les arbres d'HOGENOM qui contiennent au moins trois eucaryotes (pour être informatif sur la phylogénie de ce groupe) et au moins une bactérie (utilisée comme groupe externe pour raciner l'arbre des eucaryotes). La forêt $\mathcal{M}$ obtenue contient 794 arbres, parmi lesquels 117 sont MONO (et donc présent dans F_1) alors que 595 sont auto-cohérents (et donc résumés dans F^p_{all} par un arbre MONO). Les super-arbres obtenus en utilisant MRP et STC+*PhySIC_IST* sur ces deux forêts sont représentés dans la Figure 4.6. Pour les deux méthodes, la prise en compte des arbres MULT améliore nettement l'informativité du super-arbre produit. Dans le cas de MRP, de nombreux nouveaux clades peuvent être induits et 10 taxons supplémentaires sont présents dans le super-arbre de STC+*PhySIC_IST*.

De plus, dans le cas de STC+*PhySIC_IST*, les *Kineoplastea* (un groupe de protistes auquel appartiennent notamment les *Trypanosoma*) qui sont placés, de manière erronée, dans le même clade que les champignons (*Fungi*) et les métazoaires lorsque l'on n'utilise que F_1 retrouvent une position plus consensuelle lorsque l'on utilise F^p_{all}. L'arbre obtenu par STC+*PhySIC_IST* à partir de cette forêt est d'ailleurs globalement en accord avec nos connaissances de la taxonomie des eucaryotes (Keeling et al., 2005). Seule la position de *Caenorhabditis* (le ver nématode modèle) est discutable puisque, bien que cela fasse encore l'objet de débats, il semble néanmoins que sa position la plus probable soit plutôt comme groupe frère des insectes (Philippe et al., 2005). On peut également remarquer que l'espèce *Cyanidioschyzon merolae* (une algue rouge) qui est à la racine de l'arbre inféré avec F_1, au lieu d'être en groupe frère des *Viridiplantae*[7] (le groupe des plantes vertes) (Rodriguez-Ezpeleta et al., 2007), est exclu de l'arbre inféré avec F^p_{all}. Le fait que l'échantillonnage taxonomique pour les eucaryotes est encore assez faible explique certainement en partie ce problème. On peut espérer que l'arrivée prochaine de nombreux génomes complets permette d'enrichir cet arbre et de corriger ce problème.

7. le super-arbre inféré par STC +*PhySIC_IST* est raciné, la position de *Viridiplantae* est donc bien erronée dans cet arbre.

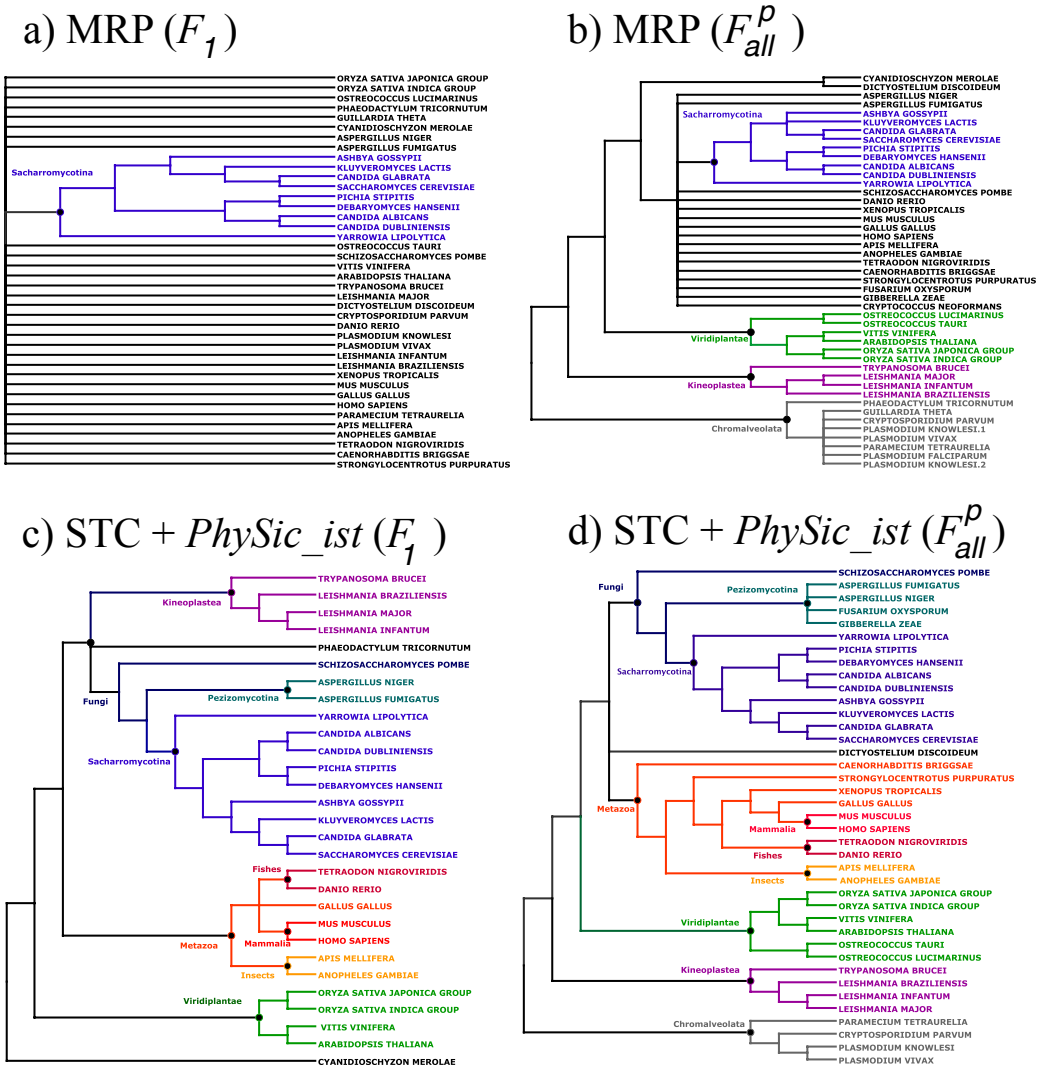

FIGURE 4.6 – **Impact de l'utilisation des arbres MULT sur le super-arbre des eucaryotes.** *Parmi les* 794 *arbres que nous avons identifiés dans* HOGENOM *pour inférer la phylogénie des eucaryotes seuls 117 sont* MONO(F_1)*. Grâce aux méthodes présentées dans cette section, on peut utiliser le signal phylogénétique de* 595 (F^p_{all}) *de ces* 794 *arbres. MRP produit un arbre nettement plus résolu si on l'utilise avec la forêt* F^p_{all} *a) que si on l'utilise avec* F_1 *b). De même, l'utilisation de* F^p_{all} *permet à STC+*PhySIC_IST *c) de proposer un super-arbre qui inclut 10 taxons de plus que si on utilise STC+*PhySIC_IST *avec* F_1 *b). L'utilisation de* F^p_{all} *permet également de corriger la position des "kineoplastea" (voir le corps du texte pour plus de détails).*

4.2 Trouver un des scénarios macro-évolutifs les plus parcimonieux

4.2.1 Motivations

Une fois que l'on dispose d'un arbre des espèces bien établi, on peut essayer de comprendre quels sont les évènements macro-évolutifs qui ont causé les différences que l'on observe entre la phylogénie des espèces et un arbre de gènes particulier. L'identification des transferts horizontaux est une question particulièrement importante, notamment parce qu'ils sont un facteur clef de la résistance des bactéries aux antibiotiques. Il semble d'ailleurs que le premier article à mentionner leur existence, paru en japonais en 1959, les ait mis en évidence en observant le transfert d'une résistance aux antibiotiques d'une bactérie à une autre (Ochiai et al., 1959). Aujourd'hui, ces transferts sont à la source des inquiétudes concernant les plantes génétiquement modifiées (e.g. Demaneche et al., 2008; Keese, 2008). Sur un plan plus théorique, ils sont également une source de débat en phylogénie. En effet, certains auteurs pensent que les transferts chez les procaryotes (et à proximité de la racine de l'Arbre de la Vie) sont si importants qu'un "réseau de la vie" serait plus approprié qu'une simple arborescence (Doolittle, 1999; Goldenfeld and Woese, 2007). Des études complémentaires semblent toutefois indiquer que les transferts n'oblitèrent pas complètement le signal évolutif de spéciation et qu'un Arbre de la Vie peut encore être discerné malgré le bruit qu'ils engendrent (Daubin et al., 2003; Kurland et al., 2003; Puigbo et al., 2009). Même si ce débat n'est pas encore clos, il a d'ores et déjà engendré des progrès considérables. Par exemple, il est désormais bien établi que la détection de transferts par une approche phylogénétique est plus fiable que par comparaisons de séquences (Kurland et al., 2003; McInerney et al., 2008; Vernot et al., 2008). L'approche phylogénétique la plus populaire est la *réconciliation d'arbres* qui se base sur une comparaison détaillée de l'arbre de gènes étudié et de l'arbre d'espèces référent. Ce dernier n'est pas toujours connu, mais il peut être estimé de manière satisfaisante par des analyses phylogénomiques basées sur les séquences moléculaires de nombreux gènes ou sur des caractéristiques de génomes complets (McInerney et al., 2008).

Les méthodes de réconciliation permettent d'expliquer les différences observées entre un arbre de gènes et un arbre d'espèces suite à des évènements de transfert, de duplication et/ou de perte. Une réconciliation d'arbres "plonge" l'arbre de gènes (représenté par des traits simples) dans l'arbre d'espèces (représenté par des tubes), et associe les nœuds internes de l'arbre de gènes à un évènement évolutif particulier (Page, 1994). La Figure 4.7 montre un exemple simple de cette représentation schématique d'une réconciliation.

Les approches pour réconcilier un arbre (MULT) de gènes (noté G) et un arbre (MONO) d'espèces (noté S) se basent sur des modèles combinatoires (Goodman et al., 1979; Page, 1994; Guigo et al., 1996; Hallett et al., 2004; K. Yu. Gorbunov, 2010) ou probabilistes (Arvestad et al., 2003; Tofigh et al., communication personnelle). Ces derniers intègrent plusieurs paramètres et offrent une meilleure représentation de

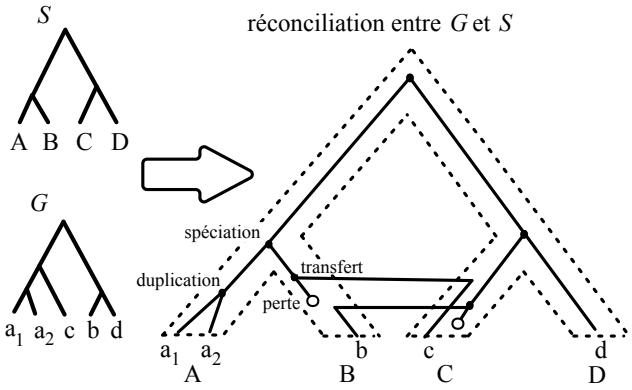

FIGURE 4.7 – **Exemple de réconciliation entre un arbre de gènes et un arbre d'espèces.** *Dans la représentation schématique de la réconciliation entre l'arbre de gènes G et l'arbre d'espèces S, la structure de l'arbre d'espèces est représentée par des "tubes" en pointillés dans lesquels est plongé l'arbre de gènes G représenté par des traits pleins. En plus des trois spéciations communes, un évènement de duplication, deux transferts et deux pertes (représentées par des $\circ$) permettent d'expliquer les différences qui existent entre ces deux arbres et donc de les "réconcilier".*

l'évolution génomique que les modèles combinatoires, mais ils sont beaucoup plus exigeants en espace mémoire et en temps de calcul. C'est pourquoi seuls les modèles combinatoires sont actuellement utilisables pour des études phylogénomiques de plusieurs milliers de familles de gènes. Cependant, avec les progrès des nouvelles technologies de séquençage, même ces modèles sont en voie de devenir trop lents. Nous avons proposé un modèle combinatoire de réconciliation qui considère les évènements de spéciation, de duplication, de transfert et de perte (respectivement notés $\mathbb{S}$, $\mathbb{D}$, $\mathbb{T}$, et $\mathbb{L}$ [8]), et un algorithme d'une complexité meilleure que les solutions existantes. Nous avons également conduit une série de simulations afin d'évaluer la pertinence du critère de parcimonie pour différents taux réalistes de duplication, de transfert et de perte de gènes.

Ces travaux ont commencé lors de la dernière année de thèse de Céline Scornavacca au cours de laquelle Céline et Vincent Berry ont effectué un bref séjour dans une équipe Russe qui travaillait sur la réconciliation d'arbres (K. Yu. Gorbunov, 2010). Après le départ de Céline, nous avons recruté Jean-Philippe Doyon pour un contrat post-doctoral de deux ans (un an au LIRMM suivi d'un an à l'ISEM) dans le cadre de l'ANR Phyl-Ariane. Ce sujet est alors devenu une de nos priorités, et Jean-Philippe, que je co-encadre avec Vincent Berry, a su faire avancer rapidement ce projet. Evidemment, Céline est restée impliquée dans ces travaux et est donc co-auteurs des publications qui en découlent. L'équipe du LBBE (partenaire Lyonnais de l'ANR) travaille, quant à elle, sur une approche probabiliste, leurs travaux sont au cœur du générateur d'histoires de gènes/histoires d'espèces que nous avons utilisé lors des simulations visant à valider l'approche par parcimonie. Les travaux décrits dans cette section vont être présentés à *Recomb CG : RECOMB Satellite Workshop*

8. La notation $\mathbb{L}$ utilisée pour les pertes, fait référence à la terminologie anglaise : *Loss*

on Comparative Genomics dont les actes sont publiés dans la série *Lecture Notes in Bioinformatics* (Doyon et al., 2010). Suite à cette présentation, il est possible que notre article soit sélectionné pour une version étendue dans le journal *Journal of Computational Biology*. Nous travaillons également à un article de synthèse sur ce domaine qu'un éditeur de la revue *Briefings in Bioinformatics* nous a invité à soumettre.

4.2.2 Une solution algorithmique efficace

Notre objectif est de résoudre le problème d'optimisation nommé *Réconciliation la Plus Parcimonieuse* (ou *MPR*[9]) : pour un arbre d'espèces S, un arbre de gènes G et des coûts associés aux évènements $\mathbb{D}$, $\mathbb{T}$, $\mathbb{L}$ et $\mathbb{S}$, trouver une réconciliation de coût minimum (où le coût d'une réconciliation est défini par la somme des coûts des évènements induits par le plongement de G dans S). Par exemple, si les coûts des évènements $\mathbb{D}$, $\mathbb{T}$, $\mathbb{L}$, et $\mathbb{S}$ sont respectivement $5, 10, 1$ et 0, alors le coût de la réconciliation représentée à la Figure 4.7 est 27 (1 $\mathbb{D}$, 2 $\mathbb{T}$, 2 $\mathbb{L}$, 3 $\mathbb{S}$). L'algorithme décrit dans cette section considère un coût de spéciation de zéro, mais il est facile de l'adapter pour permettre un coût non nul.

Lorsque G est binaire et que les évènements $\mathbb{T}$ ne sont pas considérés, il existe des solutions polynomiales au problème MPR (Vernot et al., 2008). Il existe même des algorithmes de complexité linéaire lorsque G et S sont tous les deux des arbres binaires (Zhang, 1997). Cependant, même pour de tels arbres, le problème MPR devient NP-complet, dès lors que l'on considère les transferts (Hallett et al., 2004; Tofigh et al., 2010). Ceci est directement lié au fait que les transferts induisent des contraintes chronologiques difficiles à vérifier entre les nœuds de S. En effet, comme les transferts se passent "horizontalement" entre deux espèces qui vivent au même moment, ils imposent des contraintes temporelles entre deux nœuds de S (qui ne sont pas ancêtres l'un de l'autre) qui s'ajoutent aux contraintes initiales (un nœud est nécessairement plus ancien que ses descendants). Ainsi, toute réconciliation ayant plusieurs transferts peut engendrer des contraintes temporelles qui sont mutuellement incompatibles (cf. Figure 4.8).

Plusieurs approches ont été proposées pour surmonter la difficulté liée aux contraintes temporelles induites par les transferts. Une première solution (Górecki, 2004; Hallett et al., 2004) est de définir à l'avance (par des moyens externes à la méthode de réconciliation) les paires de branches de S entre lesquelles les transferts sont autorisés. On construit ensuite un "graphe d'espèces" $S^{\mathbb{T}}$ en ajoutant à S une connexion horizontale entre chaque couple de branches de S pour lequel un transfert est autorisé. La réconciliation plonge l'arbre de gènes G non plus dans S mais dans $S^{\mathbb{T}}$ et une des réconciliations les plus parcimonieuses se calcule en temps $O(|S^{\mathbb{T}}|^3 \cdot |G|)$. Cependant, calculer un graphe d'espèces $S^{\mathbb{T}}$ induisant une réconciliation la plus parcimonieuse est un problème NP-complet (Górecki, 2004).

Une approche plus prometteuse est de considérer une variante réaliste du problème MPR où les nœuds de l'arbre d'espèces sont datés. Cette piste, initialement

9. Cet acronyme est lié à l'intitulé anglophone du problème : *Most Parsimonious Reconciliation*

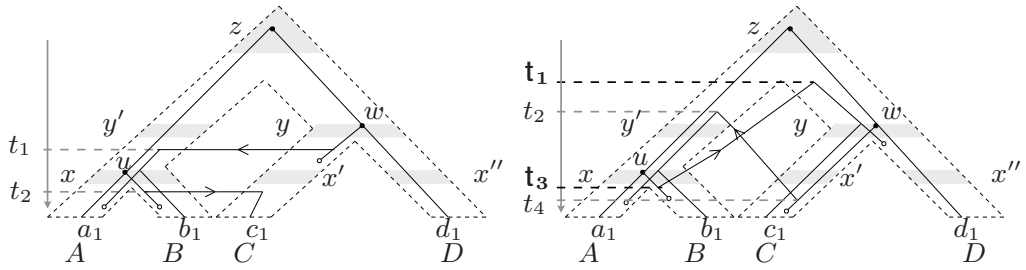

FIGURE 4.8 – **Consistance temporelle de réconciliations.** *Deux scénarios de réconciliation entre l'arbre de gènes G et l'arbre d'espèces S sont représentés suivant les mêmes conventions que dans le Figure 4.7. Le scénario de gauche est temporellement consistant ; tandis que celui de droite ne l'est pas. Les deux transferts du scénarios de droite sont mutuellement incompatibles. Suivant que u est antérieur ou postérieur à w l'un ou l'autre de ces transferts est possible (i.e. représentable par un trait horizontal), mais ils ne peuvent pas avoir eu lieux tous les deux. En effet, le transfert qui part du donneur au temps t_3 (resp. t_4) vers le receveur au temps t_1 (resp. t_2) implique que u précède (resp. succède) w.*

proposée pour des études de coévolution (Merkle and Middendorf, 2005; Conow et al., 2010; Merkle et al., 2010), est désormais reprise pour la réconciliation d'un arbre de gènes avec un arbre d'espèces (K. Yu. Gorbunov, 2010; Tofigh et al., 2010). Comme des dates relatives sont suffisantes, il n'est pas nécessaire d'utiliser des points de calibrations issues de l'étude de fossiles (Loader et al., 2007), ces données sont cependant un plus pour détecter les changements de vitesse d'évolution. La datation des nœuds de S permet d'assigner un intervalle de temps à chaque branche. Il est alors possible d'assurer la consistance individuelle de chaque transfert en vérifiant que la branche dite *donneuse* et celle dite *receveuse* ont des intervalles de temps dont l'intersection est non-vide. Si c'est le cas, le transfert respecte une contrainte locale nécessaire à la consistance temporelle de la réconciliation. La variante du problème MPR respectant cette contrainte locale peut être résolue en $O(\max(|S|,|G|)^3)$ par programmation dynamique (Merkle et al., 2010). Cependant, il est possible que des transferts soient consistants individuellement mais pas de façon conjointe, dans ce cas, la réconciliation n'est pas *globalement consistante*. Il suffit de considérer l'exemple simple de trois branches b_{01}, b_{23}, b_{03} dont les intervalles de temps relatifs sont respectivement $[0,1]$, $[2,3]$ et $[0,3]$ (plus le chiffre est grand plus on remonte dans le passé). Un transfert de b_{01} vers b_{03} est individuellement possible, tout comme un transfert de b_{03} vers b_{23}. Par contre, l'enchaînement de ces deux transferts peut conduire à un scénario qui suppose qu'une lignée de b_{01} a transféré une partie de son patrimoine génétique à une lignée de b_{23}, ce qui est impossible. De telles inconsistances peuvent être corrigées a posteriori en modifiant certains transferts (Merkle and Middendorf, 2005; Merkle et al., 2010), mais l'optimalité de la réconciliation obtenue n'est plus garantie et l'approche résultante n'est qu'une heuristique pour le problème MPR.

Une solution pour calculer une réconciliation globalement consistante optimale est de subdiviser la période couverte par S en "tranches" de temps élémentaires, et

de ne considérer que les transferts ayant un donneur et un receveur situé dans une même tranche de temps élémentaire. Cette approche permet, dans le cas d'arbres binaires, d'obtenir des algorithmes exacts pour résoudre le problème MPR comme ceux proposés par Libeskind-Hadas and Charleston (2009) et K. Yu. Gorbunov (2010). Le premier a une complexité théorique en $O(|S|^4 \cdot |G|^4)$ tandis que le second est en $O(|S|^4 \cdot k^4 \cdot |G|)$, où k est le nombre de nœuds résultants de la subdivision de S. Ces complexités, bien que polynomiales, restent élevées et impliquent des temps de calcul importants.

Certains des algorithmes décrits ci-dessus s'appuient sur un modèle combinatoire de réconciliation issu de travaux se focalisant sur les duplications et pour lesquels chaque nœud de G est *couplé* avec un seul nœud de S. Toutefois, un tel couplage est insuffisant pour les transferts car il ne peut explicitement indiquer à la fois le donneur et le receveur d'un transfert qui est immédiatement suivi d'une perte. Cette difficulté a conduit certains auteurs à ne considérer qu'une restriction du problème MPR qui néglige le coût des pertes (Hallett et al., 2004; Libeskind-Hadas and Charleston, 2009; Tofigh et al., communication personnelle).

Etant donnés un arbre de gènes G et un arbre d'espèces S daté, nous présentons un algorithme polynomial de réconciliation basé sur un modèle combinatoire où les quatre types d'évènements évolutifs ($\mathbb{DTLS}$) sont considérés. Contrairement aux approches existantes, notre algorithme gèrent correctement la combinaison d'évènements $\mathbb{T} + \mathbb{L}$. Notre modèle s'appuie sur une subdivision S' de S similaire à celle utilisée dans (K. Yu. Gorbunov, 2010; Libeskind-Hadas and Charleston, 2009; Tofigh et al., 2010) et permet de résoudre le MPR en $O(|S'| \cdot |G|)$.

4.2.2.1 Notations et définitions

Afin de pouvoir définir ce que nous entendons par "réconciliation" il est nécessaire d'introduire quelques notations supplémentaires.

Soit T un arbre enraciné, nous noterons $V(T)$ l'ensemble de ses nœuds (*Vertex*), $E(T)$ l'ensemble de ses branches (*Edge*), $r(T)$ sa racine et nous désignerons par (u, v) la branche ayant les nœuds u et v pour extrémités. Nous supposerons, par convention que la racine est en haut de l'arbre et ses feuilles en bas (ce qui oriente l'axe temporel vertical). En accord avec les notations de la section 4.1.2.1, nous continuerons de noter $\mathcal{L}(T)$ l'ensemble des feuilles de T et $L(T)$ l'ensemble de leur label. Pour un nœud u, en plus des notations T_u, u_1 et u_2 introduites dans la section 4.1.2.1 ; nous utiliserons également u_p pour désigner le nœud père de u et $T_{(u_p,u)}$ pour désigner le sous-arbre de T enraciné par la branche (u_p, u), i.e. celle au-dessus du nœud u.

Un nœud interne u de T est dit *artificiel* lorsqu'il a un seul fils, qui sera noté u_1 par convention. La *contraction* d'un nœud artificiel signifie que ce nœud est enlevé de l'arbre et que les deux branches adjacentes sont jointes, i.e : (u_p, u) et (u, u_1) sont remplacée par (u_p, u_1) dans $E(T)$. Un arbre T' est une *subdivision* d'un arbre T si l'on peut obtenir T à partir de T' par une série de contractions de nœuds artificiels.

L'arbre de gènes G est un arbre MULT qui contient un sous-ensemble des labels

utilisés dans l'arbre des espèces S ($L(G) \subseteq L(S)$). Nous noterons $L_{GS} : L(G) \to L(S)$ la fonction qui fait correspondre à chaque feuille de G l'unique feuille de S ayant le même label. Pour éviter certaines ambiguïtés, nous utiliserons le terme d'arc pour G et réserverons l'utilisation du terme de branche pour S.

Notre approche utilise un étiquetage temporel des nœuds de S où chaque feuille de S correspond à une espèce contemporaine qui existe au temps présent $t = 0$ et chaque nœud interne correspond à une espèce ancestrale qui a donné naissance à deux lignées au temps passé $t > 0$. Le temps associé à chaque nœud de S est fourni par une fonction d'étiquetage temporel.

Définition 4.2.1 – fonction d'étiquetage temporel θ :
Une fonction d'étiquetage temporel pour S, notée $\theta_S : V(S) \to \mathbb{R}$, est telle que :
i) $\theta_S(x) = 0$ pour chaque feuille $x \in L(S)$
ii) $\forall (x, x') \in V(S)^2$, si x' est un descendant de x alors $\theta_S(x') < \theta_S(x)$.

Enfin la notion de "sous-arbre homéomorphique" nous permet de faire le lien entre l'arbre G observé et l'arbre évolutif complet (incluant les pertes).

Définition 4.2.2 – sous-arbre homéomorphique $T|_K$: *Soit un arbre T et un sous-ensemble de feuilles $K \subseteq \mathcal{L}(T)$. L'arbre homéomorphique de T qui connecte K est noté $T|_K$ et représente le plus petit sous-arbre induit de T tel que $L(T|_K) = K$.*

Un scénario d'évolution valide d'un gène débute à $r(S)$ et évolue dans S par des évènements $\mathbb{DTLS}$. Cette évolution génère un arbre de gènes complet noté G^o dont les feuilles sont partitionnées en deux sous ensembles $Lc(G^o)$ et $Lp(G^o)$ selon qu'elles sont associées à des gènes contemporains (L_c) ou à des gènes perdus au cours du scénario évolutif (L_p). L'arbre de gènes G produit par ce scénario est $G^o|_{Lc(G^o)}$. La Figure 4.9 donne un exemple d'un arbre G^o et de l'arbre G correspondant. Notre définition d'un scénario $\mathbb{DTLS}$, détaillée dans (Doyon et al., 2010),

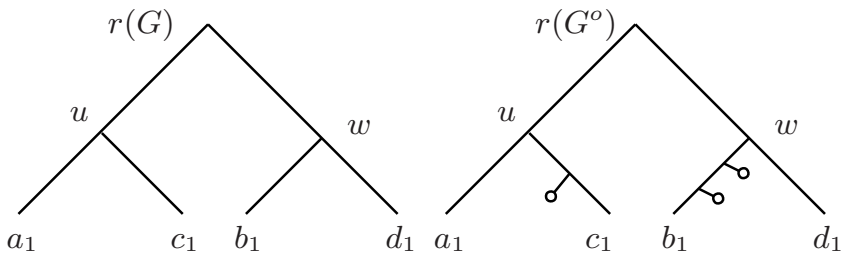

FIGURE 4.9 – **L'arbre de gènes et l'historique de ses pertes.** *Un arbre de gènes G (à gauche) peut correspondre à plusieurs scénarios évolutifs, incluant des duplications et des pertes, tels que l'arbre G^o (à droite). Si l'on restreint G^o à ses feuilles contemporaines $Lc(G^o)$ on retrouve alors l'arbre G, i.e. $G^o|_{Lc(G^o)} = G$.*

repose sur un couplage des nœuds de G^o (et non pas de G) avec ceux de S ce qui permet de caractériser précisément les transferts suivis de pertes. Le coût d'un scénario $\mathbb{DTLS}$ est simplement $d\delta + t\tau + l\lambda$, où d, t, et l dénotent respectivement le nombre

d'évènements $\mathbb{D}$, $\mathbb{T}$ et $\mathbb{L}$ de ce scénario et δ, τ et λ leurs coûts respectifs. A partir d'un scénario $\mathbb{DTLS}$ temporellement consistant on peut en générer une infinité de variantes (consistantes et de même coût) en faisant varier de manière infinitésimale la date d'un de ses évènements $\mathbb{DTLS}$. Le déplacement d'un évènement de t_1 à t_2 ne peut modifier la consistance, ou le coût, d'un scénario que s'il existe au moins un nœud de S dans l'intervalle $[t_1, t_2]$. Il existe donc des classes d'équivalences de scénarios ce qui permet de discrétiser le temps pour obtenir un modèle de réconciliation plus simple. Dans ce modèle, on considère les classes d'équivalence de scénarios et la date d'un évènement n'est pas donnée de manière précise, on indique seulement l'intervalle de temps dans lequel il se situe.

4.2.2.2 Un modèle suffisant pour trouver une des réconciliations les plus parcimonieuses

Pour obtenir un modèle efficace, l'arbre des espèces est subdivisé en tranches de temps élémentaires. Cette discrétisation du temps nous permet de trouver une des réconciliations les plus parcimonieuses en utilisant une approche basée sur la programmation dynamique.

Définition 4.2.3 – **subdivision topologique** : *Soit un arbre d'espèces (binaire) S et une fonction d'étiquetage temporel des nœuds $\theta_S : V(S) \to \mathbb{R}$. La subdivision topologique S' de S est obtenue en insérant sur chaque branche $(y_p, y) \in E(S)$ un nœud artificiel au temps $\theta_S(x)$ pour chaque nœud interne $x \in V(s)$ tel que $\theta_S(y_p) > \theta_S(x) > \theta_S(y)$.*

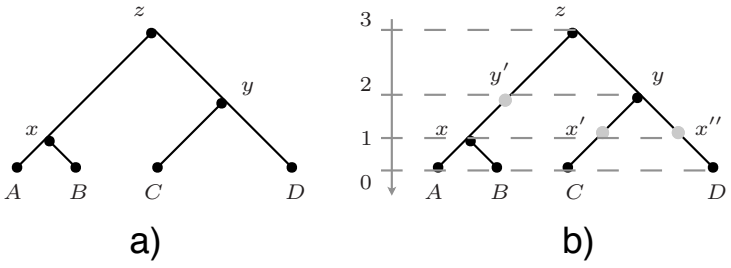

FIGURE 4.10 – **Exemple de subdivision topologique.** *Un arbre d'espèces a) et sa subdivision topologique b) pour le cas où $\theta_S(x) = 1$, $\theta_S(y) = 2$ et $\theta_S(z) = 3$. Les nœuds artificiels x', x'' et y' de cette subdivision sont en gris. L'axe vertical représente la valeur de l'étiquette temporelle des nœuds situés à ce niveau.*

On peut facilement construire une fonction d'étiquetage temporel de la subdivision topologique de S (noté S'). Il suffit de prendre comme valeur de $\theta_{S'}(x)$ le nombre de branches qui séparent x d'une des feuilles dont il est l'ancêtre (elles sont toutes à la même distance grâce à l'ajout des nœuds artificiels). Dans l'exemple de la Figure 4.10, $\theta_{S'}(x') = 1$ et $\theta_{S'}(y') = 2$. Dans la suite, nous utiliserons également la notation $\theta_{S'}(u_p, u)$ pour désigner l'intervalle de temps dans lequel se situe une branche (u_p, u)

de S'. Par convention, les branches liées aux feuilles sont dans le premier intervalle et pour toute branche de S' on a donc $\theta_{S'}(u_p, u) = \theta_{S'}(u_p)$.

Nous utilisons cette subdivision topologique pour définir un modèle permettant de trouver en un temps polynomial un des scénarios les plus parcimonieux en utilisant une approche par programmation dynamique. Une telle approche nécessite que nos évènements élémentaires permettent de passer d'un problème à un autre plus simple. C'est pour cette raison, que nous ne considérons pas le cas d'un transfert seul, mais que nous le combinons toujours à un autre évènement. Cela nous garantit qu'après chacun de nos évènements, nous nous ramenons à un problème de réconciliation qui porte sur une période plus courte de S' ou sur un ensemble plus petit d'arcs de G. Certains évènements, ou enchaînement, d'évènements ne sont jamais indispensables dans un scénario parcimonieux (avec des coûts positifs). C'est par exemple, le cas d'une duplication immédiatement suivie de la perte d'une des deux copies, ou d'une spéciation immédiatement suivie de la perte de ses deux descendants. De même, un scénario dans lequel une séquence a évolué seule à partir du temps t_x jusqu'au moment de sa perte au temps t_y, peut être remplacé par un scénario consistant, et de même coût, où la perte se produit en t_x. Ceci nous permet de nous limiter à un modèle $\mathbb{DTLS}$ simplifié pour lequel le problème MPR peut être traité par programmation dynamique et tel qu'un scénario optimal pour ce modèle est également optimal pour le modèle $\mathbb{DTLS}$ général. Nous travaillons actuellement à la rédaction d'une preuve concernant ce dernier point, mais les idées clefs montrant l'équivalence de ces deux modèles sont déjà mentionnées dans (Doyon et al., 2010). Le modèle simplifié que nous considérons est constitué de six évènements ou séquences d'évènements $\mathbb{DTLS}$ dont un évènement dit "nul" que l'on note $\varnothing$ (cf. Définition 4.2.4 et Figure 4.11).

Définition 4.2.4 – **modèle $\mathbb{DTLS}$ simplifié** : *Une réconciliation entre G et S, suivant le modèle $\mathbb{DTLS}$ simplifié, est notée α et associe chaque arc $(u_p, u) \in E(G)$ à une séquence ordonnée de branches de la subdivision S' que l'on note $\alpha(u_p, u)$. Dans cette séquence de ℓ éléments, $\alpha_i(u_p, u)$ dénote l'élément de rang i pour $1 \leq i \leq \ell$. Chaque branche $\alpha_i(u_p, u)$, dénotée ci-dessous (x_p, x), respecte une, et une seule, des contraintes suivantes (cf. Figure 4.11) :*

1. *si (x_p, x) est la dernière branche de la séquence i.e. $(x_p, x) = \alpha_\ell(u_p, u)$:*
   - *Si u est une feuille de G, alors x est l'unique feuille de S' ayant le même label, i.e. $L_{GS}(u) = x$ (Contrainte de couplage contemporain)*
   - *Sinon, un des cas ci-dessous est vérifié :*
     - $\{\alpha_1(u, u_1), \alpha_1(u, u_2)\} = \{(x, x_1), (x, x_2)\}$ $\mathbb{S}$
     - $\alpha_1(u, u_1) = \alpha_1(u, u_2) = (x_p, x)$ $\mathbb{D}$
     - $\{\alpha_1(u, u_1), \alpha_1(u, u_2)\} = \{(x_p, x), (x_p', x')\}$, *avec* $(x_p', x') \neq (x_p, x)$ *et* $\theta_{S'}(x_p', x') = \theta_{S'}(x_p, x)$ $\mathbb{T}$
2. *Si (x_p, x) n'est pas la dernière branche de la séquence alors soit :*
   - $\alpha_{i+1}(u_p, u) = (x, x_1)$ *et x est un nœud artificiel de S'* $\varnothing$
   - $\alpha_{i+1}(u_p, u) \in \{(x, x_1), (x, x_2)\}$ *et x n'est pas artificiel* $\mathbb{SL}$
   - $\alpha_{i+1}(u_p, u) \notin \{(x, x_1), (x, x_2)\}$ *et* $\theta_{S'}(\alpha_i(u_p, u)) = \theta_{S'}(\alpha_{i+1}(u_p, u))$ $\mathbb{TL}$

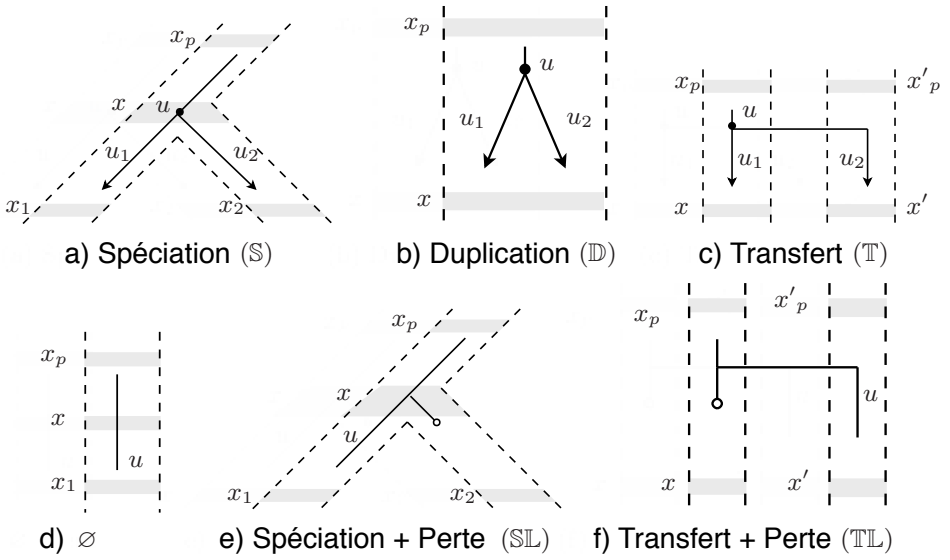

FIGURE 4.11 – **Les cas de base du modèle $\mathbb{DTLS}$ simplifié.** *Ces schémas représentent les six cas de base de la Définition 4.2.4 où un arc (u_p, u) (ligne pleine) de l'arbre G^o est plongé dans une branche (x_p, x) de S (les zones blanches des tubes en pointillés).*

4.2.2.3 Un algorithme efficace pour trouver une des réconciliations les plus parcimonieuses

Pour trouver une des réconciliations les plus parcimonieuses entre G et S, nous stockons les coûts des réconciliations entre des sous-arbres de G et des portions de S'. Afin de définir précisément ces sous-problèmes, nous utiliserons $E_t(S')$ pour désigner les branches de S' localisées au temps t et $\mathcal{F}(S', t)$ pour désigner la forêt de sous-arbres de S' enracinés bar une branche de $E_t(S')$. De manière intuitive, $\mathcal{F}(S', t)$ s'obtient à partir du dessin de S' en masquant tout ce qui est au dessus du temps t. Pour un arc $(u_p, u) \in E(G)$, une branche $(x_p, x) \in E(S')$ et un temps $t = \theta'_{S'}(x_p, x)$; nous calculons $Coût(u, x)$ qui est le coût minimal d'une réconciliation entre $G_{(u_p,u)}$ et $\mathcal{F}(S', t)$ qui utilise (x_p, x) comme première branche de la séquence $\alpha(u_p, u)$. La valeur $Coût(r(G), r(S'))$ correspond au coût minimal d'une réconciliation entre G et S'.

L'Algorithme 6 utilise le principe de la programmation dynamique (cf section 2.3.3.2) pour remplir la matrice $Coût : V(G) \times V(S') \to \mathbb{R}$ avec deux boucles imbriquées : une qui visite tous les arcs de G selon un parcours de bas-en-haut et une qui visite toutes les étiquettes temporelles de S' en remontant progressivement le temps à partir de $t = 0$. Pour un arc (u_p, u) et un temps t donnés (respectivement aux lignes 5 et 6), deux boucles consécutives sur toutes les branches $(x_p, x) \in E_t(S')$ calculent le coût minimal de l'association de (u_p, u) avec (x_p, x) selon les six évènements $\mathbb{S}$, $\mathbb{D}$, $\mathbb{T}$, $\varnothing$, $\mathbb{SL}$ et $\mathbb{TL}$ (Figure 4.11). Pour une branche $(x_p, x) \in E_t(S')$, la première boucle (lignes 11 à 21) calcule le coût minimal pour les cinq premiers évènements

tandis que la seconde boucle (lignes 22 à 25) calcule ce coût pour $\mathbb{TL}$; au final, $Coût(u,x)$ est le coût minimum associé aux six évènements. Ces calculs font appel à la fonction $MeilleurReceveur$, non détaillée ici, qui identifie le meilleur receveur pour un transfert (lignes 16, 17 et 23). $MeilleurReceveur((u,u_1),(x_p,x))$, renvoie une des branches (y_p,y) qui minimise $Coût(u_1,y)$ parmi celles de S' qui diffèrent de (x_p,x) et sont localisées au temps $\theta_{S'}(x_p,x)$.

Algorithm 6: Calcul du coût optimal d'une réconciliation entre un arbre de gènes et un arbre d'espèces

Data: Un arbre de gènes G, un arbre d'espèces S et une fonction d'étiquetage temporel θ_S.
Result: Le tableau $Coût(u,x)$ initialisé (voir le corp du texte).

1 $S' \leftarrow$ la subdivision topologique de S pour θ_S;
2 **foreach** $(u,x) \in V(G) \times V(S')$ **do**
3 **if** $u \in L(G)$ *AND* $x \in L(S')$ *AND* $L_{GS}(u) = x$ **then** $Coût(u,x) \leftarrow 0$
4 **else** $Coût(u,x) \leftarrow \infty$
5 **foreach** $(u_p,u) \in E(G)$ *selon un parcours de bas-en-haut* **do**
6 **foreach** $t \in \{0,1,\ldots,\theta_{S'}(r(S'))\}$ **do**
7 **foreach** $(x_p,x) \in E_t(S')$ **do**
8 **if** $u \in L(G)$ *AND* $x \in L(S')$ *AND* $L_{GS}(u) = x$ **then**
9 // déjà traité à l'initialisation
10 **else**
11 **foreach** $g \in \{\mathbb{S},\mathbb{D},\mathbb{T},\varnothing,\mathbb{SL}\}$ **do** $Coût_g \leftarrow \infty$
12 **if** u *a deux fils* **then**
13 **if** x *a deux fils* **then**
14 $Coût_{\mathbb{S}} \leftarrow \min \begin{cases} Coût(u_1,x_1)+Coût(u_2,x_2) \\ Coût(u_1,x_2)+Coût(u_2,x_1) \end{cases}$
15 $Coût_{\mathbb{D}} \leftarrow Coût(u_1,x)+Coût(u_2,x)+\delta$
16 $(y_p,y) \leftarrow MeilleurReceveur((u,u_1),(x_p,x))$
17 $(z_p,z) \leftarrow MeilleurReceveur((u,u_2),(x_p,x))$
18 $Coût_{\mathbb{T}} \leftarrow min \begin{cases} Coût(u_1,x)+Coût(u_2,z)+\tau \\ Coût(u_1,y)+Coût(u_2,x)\}+\tau \end{cases}$
19 **if** x *a un seul fils* **then** $Coût_{\varnothing} \leftarrow Coût(u,x_1)$
20 **if** x *a deux fils* **then** $Coût_{\mathbb{SL}} \leftarrow \min\{Coût(u,x_1),Coût(u,x_2)\}+\lambda$
21 $Coût(u,x) \leftarrow \min\{Coût_g : g \in \{\mathbb{S},\mathbb{D},\mathbb{T},\varnothing,\mathbb{SL}\}\}$
22 **foreach** $(x_p,x) \in E_t(S')$ **do**
23 $(x_p^{'},x') \leftarrow MeilleurReceveur((u_p,u),(x_p,x))$
24 $Coût_{\mathbb{TL}} \leftarrow Coût(u,x')+\tau+\lambda$
25 $Coût(u,x) \leftarrow \min\{Coût(u,x),Coût_{\mathbb{TL}}\}$
26 **return** $Coût(r(G),r(S'))$

Cet algorithme peut être implémenté en $O(|S'|.|G|)$ en optimisant légèrement l'écriture de l'Algorithme 6 que nous avons présenté ici dans sa version la plus simple (i.e. qui suit au plus près la définition du modèle $\mathbb{DTLS}$ simplifié). La preuve de cette complexité, incluant évidemment le détail des optimisations nécessaires pour l'atteindre, sont disponibles dans (Doyon et al., 2010, Annexe A). Cette annexe contient également les grandes lignes de la preuve de correction de cet algorithme (nous travaillons actuellement à la rédaction d'une preuve plus formelle). Suivant

un principe analogue à celui utilisé pour l'alignement de séquences (et détaillé à la section 2.3.3.2), on peut obtenir en $O(|S'| + |G|)$ une des réconciliations optimales en explorant la matrice $Coût(u, x)$ à partir de sa dernière case.

4.2.3 Validation de l'approche parcimonieuse par simulation

Nous avons évalué sur un grand nombre d'arbres de gènes simulés les performances de notre approche en comparant les scénarios évolutifs que nous inférons avec les vrais. Chaque paire (arbre de gènes, vrai scénario) est obtenue par un modèle probabiliste d'évolution qui inclut des duplications, des transferts et des pertes. L'Algorithme 6 est exact (il trouve toujours un des scénarios les plus parcimonieux), le but de ces simulations n'est donc pas d'évaluer ses performances, mais d'évaluer la pertinence du critère de parcimonie dans le cadre de la réconciliation d'arbres.

4.2.3.1 Protocole

Nous avons généré aléatoirement 10 arbres d'espèces contenant chacun 100 taxons sur la base d'un processus Markovien dit de *birth and death* (e.g. Yang and Rannala, 1997). Pour cela, nous avons utilisé le programme PhyloGen (Rambaut, 2002) avec un ratio de *birdth/death* fixé à 1,25. Ces arbres ont ensuite été normalisés afin qu'ils aient tous la même hauteur H. Cette hauteur correspond au temps qui s'est écoulé entre les espèces actuelles (i.e. les feuilles de l'arbre) pour lesquelles $t = 0$ et leur ancêtre commun (i.e. la racine de l'arbre) pour lequel $t = H$.

A partir d'une seule copie d'un gène, présente à la racine d'un arbre S, nous avons généré des scénarios $\mathbb{DTLS}$ en faisant évoluer cette copie selon un processus de Poisson caractérisé par trois paramètres : le taux de duplication (r_δ), le taux de transfert (r_τ) et le taux de perte (r_λ). Dans le cas d'un transfert, le donneur est choisi uniformément parmi les différents receveurs potentiels (i.e. les gènes existants au moment du transfert). On obtient ainsi, pour chaque simulation, un arbre de gènes G^o et une réconciliation simulée α_R incluant les évènements $\mathbb{DTLS}$ à l'origine de G^o.

Csuros and Miklos (2009) ont récemment étudié l'ampleur relative des taux de duplication, de transfert et de perte chez les archébactéries. Ils estiment qu'environ 23% des évènements sont des duplications, 1% sont des acquisitions (par transferts) et 76% sont des pertes. Ils observent également un taux approximatif de perte de 1.5 pour un arbre d'hauteur unitaire. En nous appuyant sur ces résultats, nous avons fait varier de manière réaliste les taux $\mathbb{D}$, $\mathbb{T}$ et $\mathbb{L}$ pour créer deux séries de jeux de tests.

Le premier jeu de données, nommé ds_1, correspond aux paramètres suivants : le taux de perte r_λ est 0.7, la hauteur des arbres d'espèces (H) est 1 et les taux r_δ et r_τ varient dans l'intervalle $[0.01, 0.35]$ avec un pas de 0.034 (soit 11 valeurs). Nous avons donc obtenu 11×11 ensembles de paramètres cohérents avec une évolution le long d'une échelle temporelle importante correspondant, par exemple, au phylum des bactéries ou à celui des archébactéries. En effet, le taux de perte choisi est

réaliste (selon Csuros and Miklos, 2009) et nous ne faisons pas de supposition sur le taux relatif de transfert et de perte, la seule contrainte étant que $r_\delta + r_\tau \leq r_\lambda$. Pour chacun des 10 arbres d'espèces et des 121 ensembles de paramètres, nous avons généré 5 arbres de gènes soit un total de 6 050 réconciliations. L'objectif est d'étudier la pertinence de la parcimonie suivant l'importance relative des différents types évènements.

Le deuxième jeu de données, nommé ds_2, fixe le rapport $r_\lambda/(r_\lambda + r_\delta + r_\tau)$ à 0.7 (Csuros and Miklos, 2009). L'objectif est ici d'étudier la pertinence d'une approche de parcimonie pour différentes échelles temporelles (phylogénies profondes ou récentes) en variant la hauteur H de S comme suit : $H = 0.2, 0.4, 0.8$ et 1.6. Le taux de transfert r_τ varie dans l'intervalle $[0, 0.3]$ avec un pas de 0.03 (soit 11 valeurs) et en imposant $r_\delta = 0.3 - r_\tau$. Pour chacun des 10 arbres d'espèces et des 44 combinaisons de paramètres, 20 arbres de gènes ont été générés soit un total de 8 800 réconciliations.

4.2.3.2 Analyse des résultats

Pour chaque jeu de données, nous avons utilisé comme coût d'un évènement $\mathbb{DTL}$ l'inverse du taux moyen de ce type d'évènements durant le processus de simulation (par exemple, δ est fixé à $1/0.18$ pour ds_1). Pour chaque couple d'arbres (G, S), nous avons inféré une des réconciliation les plus parcimonieuses, nommée α_p, grâce à une implémentation de l'Algorithme 6 disponible à l'adresse suivante : `http://www.atgc-montpellier.fr/MPR/`. Cette inférence prend en moyenne de l'ordre d'une seconde sur une machine linux standard [10].

Il faut noter qu'une réconciliation réelle α_R contient souvent des évènements qui ne concernent que des lignées éteintes et pour lesquels il n'existe aucune trace dans l'arbre de gènes. Il est donc illusoire d'espérer retrouver de tels évènements, ce qui n'est pas gênant vu qu'ils ne sont pas informatifs pour l'étude d'espèces contemporaines. C'est par exemple le cas d'évènements de duplication immédiatement suivis de pertes ou d'enchaînements d'évènements $\mathbb{TL}$. Afin de comparer α_p avec le vrai scénario évolutif α_R, nous avons éliminé de celui-ci une partie de ces évènements dits "fantômes" et ainsi obtenu une nouvelle réconciliation notée α'_R à laquelle nous nous comparons. Nous travaillons actuellement à une définition formelle de ces évènements. Pour l'instant nous n'éliminons que les plus immédiats, ce qui peut nous conduire à sous-estimer la pertinence de l'approche MPR.

Nous avons d'abord étudié les conditions dans lesquelles la parcimonie peut correctement estimer les évènements $\mathbb{DTL}$ en comparant les coûts de α_p et α'_R : quand ils diffèrent de façon importante, la parcimonie n'est plus une approche souhaitable. Le surcoût relatif de α'_R par rapport à une des réconciliations les plus parcimonieuses est défini par :

$$Surcoût(\alpha'_R, \alpha_P) \quad = \quad \frac{Coût(\alpha'_R) - Coût(\alpha_P)}{Coût(\alpha_P)}.$$

10. processeur 3GHz et 4Go de RAM

Il faut noter que $Coût(\alpha'_R) = Coût(\alpha_P)$ n'implique pas $\alpha_P = \alpha'_R$ puisqu'il peut exister plusieurs réconciliations de coût minimal. La Figure 4.12 montre les variations du surcoût en fonction du taux de délétion et de transfert (analyse de ds_1) ainsi qu'en fonction du taux de délétion et de la hauteur de l'arbre S (analyse de ds_2). On remarque que le surcoût reste très limité pour toutes les combinaisons de taux mais qu'il augmente sensiblement avec la hauteur de l'arbre. Cette tendance globale est attendue. En effet, plus un scénario contient d'évènements plus il est probable qu'une succession de ces évènements s'annule ou soit équivalente à un seul évènement et que l'explication la plus parcimonieuse ne soit donc pas la bonne.

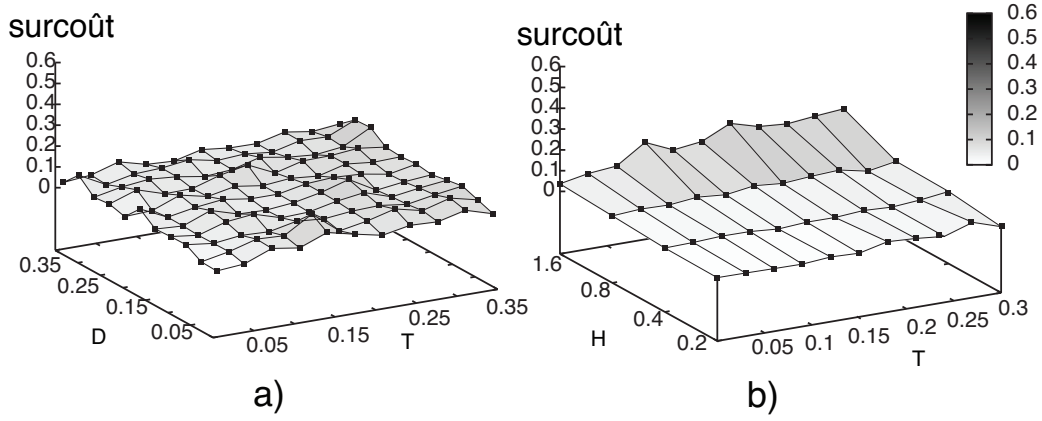

FIGURE 4.12 – **Surcoût relatif de la vrai réconciliation par rapport aux plus parcimonieuses.** *L'augmentation du surcoût de la réconciliation réelle est représentée par l'utilisation de niveaux de gris de plus en plus foncés. Ce surcoût, par rapport aux réconciliations les plus parcimonieuses, est étudié en fonction du taux de* D*uplication et de* T*ransferts a) ainsi qu'en fonction de la* H*auteur de l'arbre des espèces et du taux de* T*ransfert.*

Nous nous sommes ensuite penchés sur les conditions pour lesquelles la parcimonie retrouve correctement la position des évènements $\mathbb{DTL}$ qui ont engendré G. Rappelons qu'une réconciliation α pour un arbre de gènes G définit les évènements $\mathbb{DTL}$ associés aux nœuds et branches internes de G. Puisque la position des duplications et des transferts indique univoquement celle des pertes, nous nous sommes focalisés sur les évènements $\mathbb{D}$ et $\mathbb{T}$.

Soit $\mathbb{D}_S(\alpha)$ le sous-ensemble de paires$(u, (x_p, x)) \in V(G) \setminus L(G) \times E(S)$ tel que u est une duplication localisée sur (x_p, x) selon α. Soit $\mathbb{T}_S(\alpha)$ le sous-ensemble de triplets $((u_p, u), (x_p, x), (y_p, y)) \in E(G) \times E(S)^2$ tel que la branche (u_p, u) de l'arbre de gène est impliquée dans un transfert allant de la branche (x_p, x) de l'arbre des espèces vers (y_p, y). Pour une réconciliation parcimonieuse α_P, la précision avec laquelle elle retrouve les évènements $\mathbb{D}$ et $\mathbb{T}$ de la réconciliation réelle est évaluée

par les ratios de faux positifs/négatifs définis ci-dessous (où $\mathbb{E} \in \{\mathbb{D}, \mathbb{T}\}$).

$$FP_{\mathbb{E}}(\alpha'_R, \alpha_P) = \frac{|\mathbb{E}_S(\alpha_P) - \mathbb{E}_S(\alpha'_R)|}{|\mathbb{E}_S(\alpha_P)|}$$
$$FN_{\mathbb{E}}(\alpha'_R, \alpha_P) = \frac{|\mathbb{E}_S(\alpha'_R) - \mathbb{E}_S(\alpha_P)|}{|\mathbb{E}_S(\alpha'_R)|}$$

Les Figure 4.13 et Figure 4.14 montrent l'évolution de ces ratios en fonction de la hauteur de l'arbre S et des taux de duplication et de transfert. La Figure 4.13 montre que $FP_{\mathbb{D}}$ est proche de zéro pour toutes les conditions testées : quasiment toutes les duplications inférées par notre algorithme sont présentes dans α'_R. Les valeurs nettement plus élevées de $FN_{\mathbb{D}}$ peuvent avoir plusieurs causes :

1. α'_R peut contenir des duplications n'ayant pas laissé de traces et donc impossibles à retrouver, i.e. des évènements fantômes non détectés ;
2. l'utilisation d'un coût de duplication proche, ou supérieur, à celui des transferts peut amener à inférer un évènement de type $\mathbb{T}$ au lieu de $\mathbb{D}$ (ce qui expliquerait aussi le taux élevé de $FP_{\mathbb{T}}$) observé dans la Figure 4.14) ;
3. il est possible qu'il existe de nombreuses réconciliations également parsimonieuses et dans ce cas le critère de parcimonie n'est pas discriminant. En prenant aléatoirement une des réconciliations les plus parcimonieuses nous prenons évidemment le risque d'inférer des duplications erronées. Une telle situation expliquerait également le taux élevé de $FN_{\mathbb{T}}$).

Malgré ces réserves concernant les taux de faux négatifs, l'aptitude de la parcimonie à identifier correctement les évènements macro-évolutifs est globalement satisfaisante, surtout pour des valeurs de H relativement faibles (i.e. pour des histoires évolutives qui ne sont pas trop anciennes).

4.3 Conclusion et perspectives

Bien qu'elles constituent la majorité des familles de gènes, les familles multigéniques sont généralement ignorées des analyses phylogénomiques. Après avoir défini la notion de nœud de duplication observable, nous avons formalisé les propriétés d'auto-isomorphisme et d'auto-cohérence. Ces propriétés permettent de s'assurer que le signal de spéciation d'un arbre MULT n'est pas contradictoire suivant les copies que l'on considère (auto-cohérence) et qu'il peut être représenté de manière exacte par un arbre MONO (auto-isomorphisme). A partir de ces propriétés, nous avons développé une série d'algorithmes et d'outils qui permettent, pour la première fois, d'intégrer le signal phylogénétique d'arbres MULT dans une analyse de type super-arbre. L'application de ces outils sur des données issues d'HOGENOM confirme que les familles multigéniques représentent une part importante du signal phylogénétique disponible et que l'utilisation de leurs arbres de gènes permet d'inférer des super-arbres de meilleures qualités. Ces résultats sont très encourageants et nous comptons poursuivre notre effort sur l'exploitation de ces arbres MULT. En

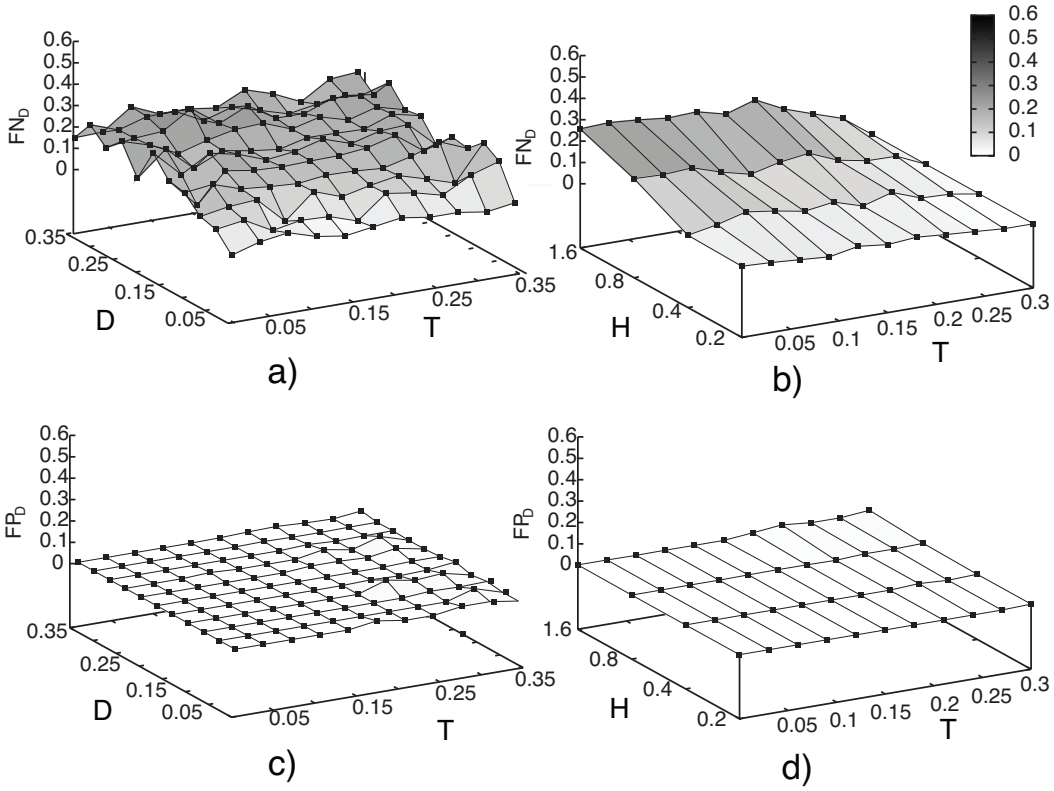

FIGURE 4.13 – **Aptitude de la parcimonie à retrouver précisément les duplications de gènes.** *Ratios de faux négatifs (a-b) et faux positifs (c-d) pour les évènements de type duplication en fonction du taux de* Duplication *et de* Transfert *a) et c) ainsi qu'en fonction de la* Hauteur *de l'arbre des espèces et du taux de* Transfert *b) et d).*

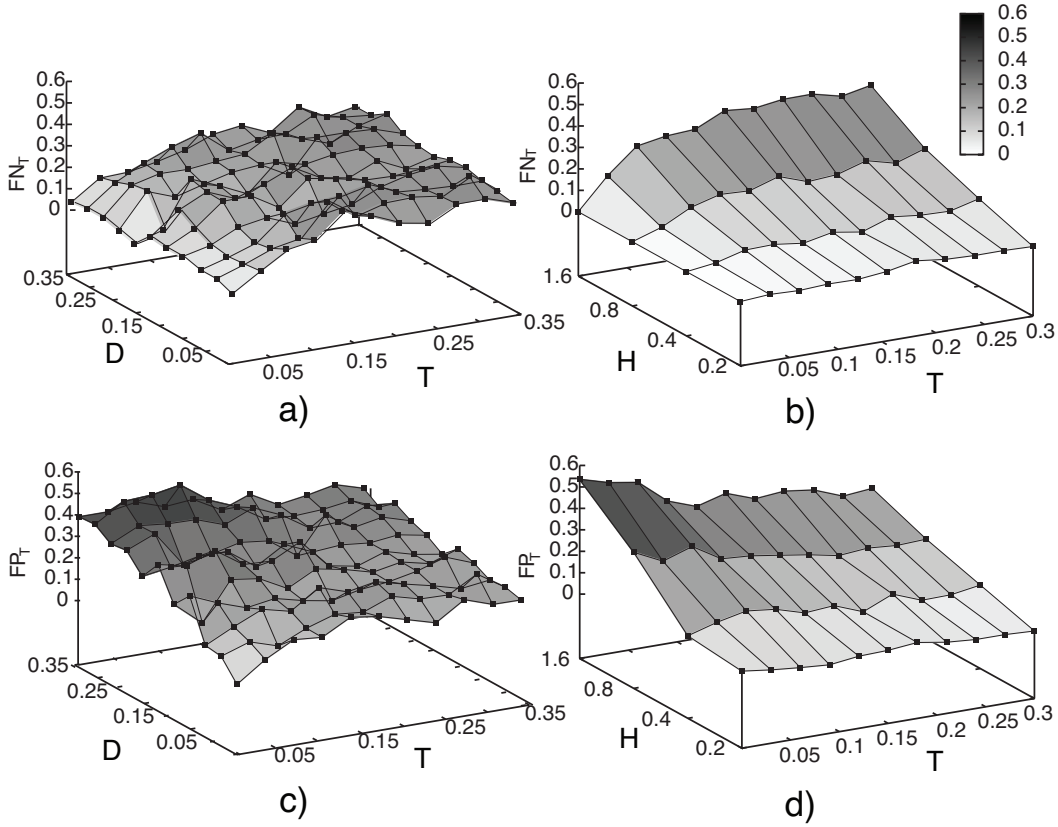

FIGURE 4.14 – **Aptitude de la parcimonie à retrouver précisément les transferts de gènes.** *Ratios de faux négatifs (a-b) et faux positifs (c-d) pour les évènements de type transfert en fonction du taux de* Duplication *et de* Transfert *a) et c) ainsi qu'en fonction de la* Hauteur *de l'arbre des espèces et du taux de* Transfert *b) et d).*

effet, nous sommes pleinement conscients des limites de nos solutions actuelles. Nous réfléchissons notamment à une manière de prendre en compte la fiabilité des clades des arbres de gènes et à une identification des "nœuds de transfert apparent" pour pouvoir les traiter de manière différente des "nœuds de duplication apparente".

Nous avons également développé un algorithme polynomial exact pour résoudre le problème MPR suivant un modèle $\mathbb{DTLS}$ respectant les contraintes temporelles liées aux transferts. Nos simulations indiquent que l'utilisation du principe de parcimonie est pertinente pour inférer les évènements macro-évolutifs qui engendrent des différences de topologies entre un arbre de gènes et un arbre d'espèces. Ces simulations sont, à notre connaissance, les premières de ce type. Elles soulèvent de nombreuses questions théoriques et pratiques concernant notamment la manière de mesurer la (di)similarité entre deux réconciliations ou la façon de caractériser les "évènements fantômes" d'un scénario simulé (i.e. ceux qui ne concernent que des lignées disparues). Nous comptons adapter notre algorithme au cas d'arbres non-binaires ou, à défaut, prouver que la réconciliation de tels arbres reste un problème NP-complet même lorsque l'arbre des espèces est daté. Une autre extension de nos travaux consiste à généraliser le modèle $\mathbb{DTLS}$ afin d'attribuer un coup plus faible aux transferts (ou aux duplications) de plusieurs gènes localisés sur une même portion du chromosome. Il semble biologiquement raisonnable d'utiliser une fonction de coût affine suivant le même principe que la fonction de pénalité des gaps dans l'alignement de séquences : il est plus probable que le transfert de trois gènes voisins sur le chromosome se soit fait en une fois plutôt qu'en trois. Ceci dit, pour pouvoir appliquer une telle fonction de coût, il faut non seulement connaître les cartes chromosomiques des espèces actuelles, mais également pouvoir inférer celles des espèces ancestrales.

A terme, il est possible que les deux axes de recherche présentés dans ce chapitre fusionnent. Nous réfléchissons notamment à une méthode de super-arbre capable de gérer des arbres sources qui soient MULT. Un critère possible est alors de chercher le super-arbre tel que la somme des coûts de réconciliation entre lui et les arbres sources soit minimale. Une des limites d'une telle approche, qui de manière plus générale est également une limite des méthodes de réconciliation, est que toute différence entre l'arbre de gènes et l'arbre des espèces est expliquée par des évènements macro-évolutifs de type $\mathbb{DTLS}$. Or ces différences sont souvent le reflet de problèmes méthodologiques (attraction des longues branches, biais de composition des séquences, modèle d'évolution des séquences inadapté). Une approche alternative pourrait consister à utiliser tour à tour les deux approches présentées dans ce chapitre. On commencerait alors par inférer une première version de l'arbre des espèces T_1 en utilisant STC $+PhySIC\_IST$ sur une forêt $\mathcal{F}_1$ contenant les arbres MONO dérivés d'arbres auto-cohérents. Puis, on utiliserait l'approche MPR sur T_1 pour inférer l'ensemble R_1 contenant les réconciliations de T_1 avec chaque arbre de gènes. R_1 fournirait alors une annotation des nœuds internes des arbres sources plus précise que celle reposant sur les nœuds de duplication observée. On peut donc raisonnablement espérer pouvoir déduire de R_1 une meilleure forêt $\mathcal{F}_2$ permettant d'inférer un meilleur super-arbre T_2 et donc de meilleures réconciliations R_2 etc.

L'exploration de ces différents axes de recherches sera au cœur des travaux que Jean-Philippe Doyon va mener à l'ISEM au cours de l'année à venir et du sujet de thèse, financé par le gouvernement vietnamien, que Nguyen Hau devrait démarrer en octobre prochain. Dans les deux cas, il s'agit d'un co-encadrement que j'effectue en partenariat avec V. Berry.

Bien qu'il existe de nombreuses améliorations possibles des méthodes décrites dans ce chapitre, elles nous permettent d'ores et déjà d'inférer automatiquement une partie conséquente de l'Arbre de la Vie et de proposer une réconciliation raisonnable entre cet Arbre de la Vie et les arbres de gènes utilisés pour l'inférer. Nous pensons conduire prochainement ce travail à l'échelle HOGENOM et mettre les résultats obtenus à disposition sur un serveur Web. Ce serveur doit permettre à la fois de naviguer dans l'Arbre de la Vie mais également de faire des requêtes basés sur les réconciliations inférées à partir de cet arbre. L'idée et de pouvoir déterminer, par exemple, l'existence de lien entre le nombre de transferts d'une famille de gènes et les processus biologiques dans lesquels elle est impliquée ; ou de détecter des groupes taxonomiques qui sont receveurs de nombreux transferts mais pas donneurs (ou le contraire). Afin d'évaluer la faisabilité d'un tel projet, nous avons déjà commencé à travailler sur une maquette de ce serveur Web. Cette maquette, réalisée par Jean-François Dufayard (recruté en CDD dans le cadre de l'ANR Phyl-Ariane), est visible sur le site http://www.atgc-montpellier.fr/reconciliation_bank/.

Chapitre 5

Conclusion

La grandeur d'un métier est, peut-être et avant tout, d'unir les hommes. Il n'est qu'un luxe véritable et c'est celui des relations humaines.

Antoine de Saint-Exupéry – Terre des hommes

Les travaux présentés dans ce manuscrit s'inscrivent dans une perspective à long terme où l'objectif est d'obtenir, à partir des génomes actuels, un Arbre de la Vie fiable, complet et annoté. Depuis mon recrutement, ce fil rouge a guidé l'essentiel de mes recherches. La constitution d'une base de données d'alignements de séquences orthologues et le développement d'une méthode d'alignement dédiée à ces séquences codantes sont des étapes préliminaires qui conditionnent la qualité des analyses ultérieures. Notre travail sur la définition de propriétés théoriques souhaitables pour assembler des phylogénies partielles en un Arbre de la Vie fiable et le développement d'outils respectant ces propriétés constituent une part centrale de ce projet. En effet, c'est avec ces outils que nous assemblons les forêts d'arbres mono-labels qui, grâce à nos travaux récents, peuvent être dérivés des familles (multi)géniques. En prenant en compte pour la première fois ces données nous répondons à une critique majeure qui pesait sur l'inférence d'un Arbre de la Vie. Enfin, nos travaux actuels sur la réconciliation d'arbres permettent d'annoter cet Arbre de la Vie en indiquant les évènements macro-évolutifs qui ont eu lieu en différents endroits de cet arbre.

Malgré les avancés décrites dans les chapitres précédents, ce manuscrit n'est évidemment pas la synthèse d'un travail achevé mais plutôt l'état de travaux en cours qui convergent vers un but en partie utopique. Comme je l'ai expliqué en introduction l'inférence exacte et complète de l'Arbre de la Vie est impossible pour de nombreuses raisons, mais cela n'empêche pas d'essayer de s'en approcher. Ce faisant nous obtenons une vision, certes imparfaite, mais toujours plus précise de cet arbre. Il est certain que les évènements de type macro-évolutifs doivent être pris en compte lors de l'inférence de l'Arbre de la Vie. Des annotations concernant les habitats et les traits d'histoires de vies des espèces peuvent nous permettre d'associer des coûts variables aux transferts ou du moins de détecter a posteriori des transferts suspects (e.g. entre une espèce marine et une espèce terrestre). Inversement, des tansferts entre deux lignées ancestrales peuvent nous renseigner sur l'une si l'on dispose d'information sur l'autre (e.g. au travers de fossiles). On peut également se demander si les gènes impliqués dans une même fonction biologique sont plus souvent transférés/dupliqués simultanément.

Afin de tirer le meilleur parti de l'ensemble des annotations associées aux arbres de gènes et à l'Arbre de la Vie, il est nécessaire de disposer de puissants outils

de requêtage. En effet, le but n'est pas simplement de produire ces données mais de pouvoir les exploiter pour mieux comprendre les processus qui façonnent l'évolution des génomes et les liens qui peuvent exister entre les fonctions des gènes et leur évolution. La *Gene Ontology* constitue une référence concernant l'annotation des gènes (Ashburner et al., 2000). Elle définit et hiérarchise l'ensemble des concepts qui peuvent apparaître dans leurs annotations fonctionnelles. En exploitant cette hiérarchie (représentée informatiquement par un graphe) on peut, lors d'une recherche, identifier des gènes pertinents même s'ils ne sont pas annotés par des termes exacts de la requête. On peut adapter ce principe pour trouver les familles de gènes ayant un représentant taxonomique proche de celui demandé ; dans ce cas, on utilise l'Arbre de la Vie comme hiérarchie. Je co-encadre depuis quelques mois la thèse de Mohameth François Sy sur l'utilisation d'ontologies pour améliorer les outils de requêtages. Bien que cette thèse ne soit pas spécifiquement faites dans un cadre bio-informatique, les résultats obtenus s'applique naturellement à ce domaine. Le fait que ma femme soit l'encadrante principale de Mohameth facilite évidemment cette collaboration.

Il est clair que les nouvelles technologies de séquençage sont un progrès majeur qui va changer l'échelle de nos analyses phylogénomiques dans un avenir proche. Mais ce changement n'est pas le seul qui risque de bousculer notre approche de l'Arbre de la Vie. En effet, l'équipe de Craig Venter vient, une nouvelle fois, de faire la une des journaux en engendrant une colonie de bactéries ayant un génome artificiel (Gibson et al., 2010). L'objectif final de leur projet étant de pouvoir créer massivement des biocarburants, on peut supposer que les choses ne vont pas s'arrêter là. Il ne s'agit d'ailleurs pas d'un phénomène isolé et l'on assiste depuis quelques années à l'émergence rapide d'une nouvelle discipline, la "biologie synthétique". L'activité centrale de cette discipline est de produire, cataloguer et assembler des fragments de séquences codantes qui constituent les briques élémentaires des futurs génomes de synthèses. Des milliers de ces fragments sont déjà répertoriés comme autant de pièces d'un gigantesque jeu de construction (e.g. `http://bbf.openwetware.org/`). On peut alors se demander comment modéliser les parentés qui existent entre ces (futurs) génomes de synthèse et les génomes "naturels", comment intégrer ces nouvelles entités dans un Arbre de la Vie et s'il est possible de prédire leur propension à être impliqués dans des évènements de transfert.

J'arrive maintenant à la fin de ce manuscrit, j'avoue avoir envisagé sa rédaction comme une corvée obligatoire, je m'étais trompé. Sa rédaction m'a permis de faire le point sur l'avancée de mes travaux et de prendre du recul. Ce n'est d'ailleurs pas par pur masochisme que j'ai rédigé près de 130 pages mais parce que j'y ai finalement pris un réel plaisir. Après cette trêve, consacré à la rédaction, je vais reprendre un rythme plus normal et sortir de l'isolement partiel dans lequel je m'étais confiné ces dernières semaines. C'est avec plaisir que je vais retrouver l'ensemble de mes collègues pour partager avec eux certaines idées et réflexions qui me sont venues lors de la rédaction de ce manuscrit. Je suis conscient de la chance que j'ai d'effectuer ce métier qui me plait et d'entretenir des collaborations fructueuses et agréables tant avec des biologistes qu'avec des informaticiens.

Table des figures

Liste des tableaux

Bibliographie

Abascal, F., R. Zardoya, and M. J. Telford. 2010. **TranslatorX : multiple alignment of nucleotide sequences guided by amino acid translations**. Nucleic Acids Res. . 38, 51

Aho, A. V., Y. Sagiv, T. G. Szymanski, and J. D. Ullman. 1981. **Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions**. SIAM J. Comp. 10 :405–421. 60, 69

Altschul, S. F. 1989. **Gap costs for multiple sequence alignment**. J. Theor. Biol. 138 :297–309. 50

Altschul, S. F. and B. W. Erickson. 1986. **Optimal sequence alignment using affine gap costs**. Bull Math Biol 48 :603–16. 44

Altschul, S. F., W. Gish, W. Miller, E. W. Myers, and D. J. Lipmanl. 1990. **Basic local alignment search tool**. J. Mol. Biol. 215 :403–410. 55

Anisimova, M. and C. Kosiol. 2009. **Investigating protein-coding sequence evolution with probabilistic codon substitution models**. Mol Biol Evol 26 :255–71. 24

Arvestad, L., A. C. Berglund, J. Lagergren, and B. Sennblad. 2003. **Bayesian gene/species tree reconciliation and orthology analysis using MCMC**. Bioinformatics 19 Suppl 1 :7–15. 106

Ashburner, M., C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. 2000. **Gene ontology : tool for the unification of biology. the Ògene ontology consortium**. Nat Genet 25 :25–9. 126

Bandelt, H.-J. and A. Dress. 1986. **Reconstructing the shape of a tree from observed dissimilarity data**. Adv in appl math 7 :309–343. 65

Bapteste, E., E. Susko, J. Leigh, D. MacLeod, R. L. Charlebois, and W. F. Doolittle. 2005. **Do orthologous gene phylogenies really support tree-thinking ?** BMC Evol Biol 5 :33. 19

Bapteste, E., E. Susko, J. Leigh, I. Ruiz-Trillo, J. Bucknam, and W. F. Doolittle. 2008. **Alternative methods for concatenation of core genes indicate a lack of resolution in deep nodes of the prokaryotic phylogeny**. Mol Biol Evol 25 :83–91. 20, 92

Baum, B. R. 1992. **Combining trees as a way of combining data sets for phylogenetic inference, and the desirability of combining gene trees**. Taxon 41 :3–10. 61

Baum, B. R. and M. A. Ragan. 2004. **The MRP method**. Pages 17–34 *in* Phylogenetic supertrees : combining information to reveal the Tree of Life (O. Bininda-Emonds, ed.) vol. 4. Kluwer. 61

Belle, E. M., L. Duret, N. Galtier, and A. Eyre-Walker. 2004. **The decline of isochores in mammals : an assessment of the GC content variation along the mammalian phylogeny**. J Mol Evol 58 :653–60. 33

Bernardi, G., B. Olofsson, J. Filipski, M. Zerial, J. Salinas, G. Cuny, M. Meunier-Rotival, and F. Rodier. 1985. **The mosaic genome of warm-blooded vertebrates**. Science 228 :953–8. 25

Berry, V. and F. Nicolas. 2004. **Maximum agreement and compatible supertrees**. Pages 205–219 *in* Proceedings of CPM (S. C. Sahinalp, S. Muthukrishnan, and U. Dogrusoz, eds.) vol. 3109 of *LNCS*. 61

Berry, V. and F. Nicolas. 2006. **Improved parameterized complexity of the maximum agreement subtree and maximum compatible tree problems**. IEEE/ACM Trans. Comput. Biol. Bioinformatics 3 :289–302. 96

Bininda-Emonds, O. R. 2005. **transAlign : using amino acids to facilitate the multiple alignment of protein-coding DNA sequences**. BMC Bioinformatics 6 :156. 38

Bininda-Emonds, O. R., D. P. Vazquez, and L. L. Manne. 2000. **The calculus of biodiversity : integrating phylogeny and conservation**. Trends Ecol Evol 15 :92–94. 2

Bininda-Emonds, O. R. P. 2004. **Phylogenetic supertrees (combining information to reveal the Tree of Life)** vol. 4 of *computational biology series*. Kluwer academic publishers. 60, 61, 66

Birdsell, J. A. 2002. **Integrating genomics, bioinformatics, and classical genetics to study the effects of recombination on genome evolution**. Mol Biol Evol 19 :1181–97. 26

Blanquart, S., A. Criscuolo, C. Douady, E. J. Douzery, N. Lartillot, H. Philippe, and V. Ranwez. 2010. **Phylogénie moléculaire**. *in* Biologie Evolutive (M. Raymond, T. Lefèvre, and F. Thomas, eds.). De Boeck. 10, 25

Bradley, R. K., A. Roberts, M. Smoot, S. Juvekar, J. Do, C. Dewey, I. Holmes, and L. Pachter. 2009. **Fast statistical alignment**. PLoS Comput. Biol. 5 :e1000392. 41

Brown, T. C. and J. Jiricny. 1989. **Repair of base-base mismatches in simian and human cells**. Genome 31 :578–83. 26

Bruen, T. C. and D. Bryant. 2008. **Parsimony via consensus**. Syst Biol 57 :251–6. 63

Bryant, D. 1997. **Building Trees, Hunting for Trees, and Comparing Trees : theory and method in phylogenetic analysis.** Ph.D. thesis University of Canterbury. 74

Bryant, D. 2002. **A classification of consensus methods for phylogenies**. Pages 163–184 *in* Bioconsensus (M. Janowitz, F.-J. Lapointe, F. McMorris, B. Mirkin, and F. Roberts, eds.) DIMACS. AMS. 61

Bryant, D. and M. Steel. 1995. **Extension operations on sets of leaf-labelled trees**. Adv. Appl. Math. 16 :425–453. 65

Carroll, H., W. Beckstead, T. O'Connor, M. Ebbert, M. Clement, Q. Snell, and D. McClellan. 2007. **DNA reference alignment benchmarks based on tertiary structure of encoded proteins**. Bioinformatics 23 :2648–2649. 42, 51

Carvalho, P., J. A. F. Diniz-Filho, and L. M. Bini. 2005. **The impact of felsenstein's "phylogenies and the comparative method" on evolutionary biology.** Scientometrics 62 :53–66. 2

Chao, K. M., R. C. Hardison, and W. Miller. 1994. **Recent developments in linear-space alignment methods : a survey**. J Comput Biol 1 :271–91. 57

Chapman, A. D. 2009. **Numbers of living species in australia and the world**. Tech. rep. Australian Biodiversity Information Services. 16

Chen, F., A. J. Mackey, J. K. Vermunt, and D. S. Roos. 2007a. **Assessing performance of orthology detection strategies applied to eukaryotic genomes**. PLoS One 2 :e383. 29

Chen, J. M., D. N. Cooper, N. Chuzhanova, C. Ferec, and G. P. Patrinos. 2007b. **Gene conversion : mechanisms, evolution and human disease**. Nat Rev Genet 8 :762–75. 26

Chevreux, B., T. Pfisterer, B. Drescher, A. Driesel, W. Müller, T. Wetter, and S. Suhai. 2004. **Using the miraEST assembler for reliable and automated mRNA transcript assembly and SNP detection in sequenced ESTs**. Genome Res. 14 :1147–1159. 55

Chow, L. T., R. E. Gelinas, T. R. Broker, and R. J. Roberts. 1977. **An amazing sequence arrangement at the 5' ends of adenovirus 2 messenger RNA**. Cell 12 :1–8. 24

Conow, C., D. Fielder, Y. Ovadia, and R. Libeskind-Hadas. 2010. **Jane : a new tool for the cophylogeny reconstruction problem**. Algorithms Mol Biol 5 :16. 109

Crick, F. 1970. **Central dogma of molecular biology**. Nature 227 :561–3. 22

Crick, F. H. 1958. **On protein synthesis**. Symp Soc Exp Biol 12 :138–63. 22

Criscuolo, A., V. Berry, E. J. Douzery, and O. Gascuel. 2006. **SDM : a fast distance-based approach for (super) tree building in phylogenomics**. Syst Biol 55 :740–55. 32, 77

Csuros, M. and I. Miklos. 2009. **Streamlining and large ancestral genomes in Archaea inferred with a phylogenetic birth-and-death model**. Mol Biol Evol 26 :2087–2095. 116, 117

Dagan, T. and W. Martin. 2006. **The tree of one percent**. Genome Biol 7 :118. 19, 92

Darwin, C. 1859. **On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life**. London. 8

Daubin, V., N. A. Moran, and H. Ochman. 2003. **Phylogenetics and the cohesion of bacterial genomes**. Science 301 :829–832. 106

Dawkins, R. 1976. **The selfish gene**. Oxford University Press, New-York. 26

Dayhoff, M., R. Schwartz, and B. Orcutt. 1978. **A model of evolutionary change in proteins.** *in* Atlas of Protein Sequence and Structure (M. Dayhoff, ed.) vol. 5. Washington, DC. 43

Dekker, M. C. 1986. **Reconstruction methods for derivation trees**. Master's thesis University of Amsterdam. 65

Delsuc, F., G. Tsagkogeorga, N. Lartillot, and H. Philippe. 2008. **Additional molecular support for the new chordate phylogeny**. Genesis 46 :592–604. 55

Demaneche, S., H. Sanguin, J. Pote, E. Navarro, D. Bernillon, P. Mavingui, W. Wildi, T. M. Vogel, and P. Simonet. 2008. **Antibiotic-resistant soil bacteria in transgenic plant fields**. Proc Natl Acad Sci U S A 105 :3957–62. 106

Doolittle, W. F. 1999. **Phylogenetic classification and the universal tree**. Science 284 :2124–2129. 106

Doolittle, W. F. and E. Bapteste. 2007. **Pattern pluralism and the Tree of Life hypothesis**. Proc Natl Acad Sci U S A 104 :2043–9. 19

Douzery, E. J. and H. Philippe. 1994. **The pitfalls of molecular phylogeny based on four species, as illustrated by the Cetacea/Artiodactyla relationships**. Journal of Mammalian Evolution 2 :133–52. 18

Douzery, E. J., J. Romiguier, K. Belkhir, F. Delsuc, N. Galtier, and V. Ranwez. soumis. **The OrthoMaM 6 database of mammalian orthologues : comparative genomics of exons and coding sequences**. Nucleic Acids Res. . 29

Doyon, J., C. , Scornavacca, K. Gorbunov, G. Szöllösi, V. Ranwez, and V. Berry. 2010. **An efficient algorithm for gene/species trees parsimonious reconciliation with losses, duplications and transfers**. *in* RECOMB CG 2010 : Proceedings of the Eighth Annual RECOMB Satellite Workshop on Comparative Genomics Lecture Notes In Bioinformatics Springer-Verlag. 92, 108, 111, 113, 115

Duret, L., M. Semon, G. Piganeau, D. Mouchiroud, and N. Galtier. 2002. **Vanishing GC-rich isochores in mammalian genomes**. Genetics 162 :1837–47. 26, 33

Dutheil, J., S. Gaillard, E. Bazin, S. Glemin, V. Ranwez, N. Galtier, and K. Belkhir. 2006. **Bio++ : a set of C++ libraries for sequence analysis, phylogenetics, molecular evolution and population genetics.** BMC Bioinformatics. 7 :188. 71, 92

Edgar, R. C. 2004a. **Local homology recognition and distance measures in linear time using compressed amino acid alphabets**. Nucleic Acids Res. 32 :380–385. 51

Edgar, R. C. 2004b. **MUSCLE : a multiple sequence alignment method with reduced time and space complexity**. BMC Bioinformatics 5 :113. 41, 50, 51

Escobar, J. S., C. Scornavacca, A. Cenci, C. Guilhaumon4, S. Santoni, E. J. Douzery, V. Ranwez, S. glémin, and J. David. soumis. **Multigenic phylogeny and analysis of tree incongruence in Triticeae (Poaceae)**. Syst Biol . 60, 81

Eulenstein, O., D. Chen, G. Burleigh, D. Fernandez-Baca, and M. J. Sanderson. 2004. **Performance of flip supertree construction with a heuristic algorithm**. Syst. Biol. 53 :299–308. 77

Eyre-Walker, A. 1993. **Recombination and mammalian genome evolution**. Proc Biol Sci 252 :237–43. 26, 27

Faith, D. P. 1992. **Conservation evaluation and phylogenetic diversity**. Biological Conservation 61 :1 – 10. 2

Felsenstein, J. 1985. **Phylogenies and the comparative method**. The American Naturalist 125 :1. 2

Fiers, W., R. Contreras, F. Duerinck, G. Haegeman, D. Iserentant, J. Merregaert, W. Min Jou, F. Molemans, A. Raeymaekers, A. Van den Berghe, G. Volckaert, and M. Ysebaert. 1976. **Complete nucleotide sequence of bacteriophage MS2 RNA : primary and secondary structure of the replicase gene**. Nature 260 :500–7. 11

Fleischmann, R. D., M. D. Adams, O. White, R. A. Clayton, E. F. Kirkness, A. R. Kerlavage, C. J. Bult, J. F. Tomb, B. A. Dougherty, J. M. Merrick, and et al. 1995. **Whole-genome random sequencing and assembly of haemophilus influenzae Rd**. Science 269 :496–512. 13

Forest, F., R. Grenyer, M. Rouget, T. J. Davies, R. M. Cowling, D. P. Faith, A. Balmford, J. C. Manning, S. Proches, M. van der Bank, G. Reeves, T. A. Hedderson, and V. Savolainen. 2007. **Preserving the evolutionary potential of floras in biodiversity hotspots**. Nature 445 :757–60. 2

Galtier, N. 2007. **A model of horizontal gene transfer and the bacterial phylogeny problem**. Syst Biol 56 :633–42. 19

Galtier, N. and V. Daubin. 2008. **Dealing with incongruence in phylogenomic analyses**. Philos Trans R Soc Lond B Biol Sci 363 :4023–9. 19

Galtier, N. and L. Duret. 2007. **Adaptation or biased gene conversion ? extending the null hypothesis of molecular evolution**. Trends Genet 23 :273–7. 37

Galtier, N., L. Duret, S. Glemin, and V. Ranwez. 2009. **GC-biased gene conversion promotes the fixation of deleterious amino acid changes in primates**. Trends Genet 25 :1–5. 21, 33

Galtier, N. and D. Mouchiroud. 1998. **Isochore evolution in mammals : a human-like ancestral structure**. Genetics 150 :1577–84. 33

Gamma, E., R. Helm, R. Johnson, and J. Vlissides. 1995. **Design Patterns**. Addison-Wesley, Boston, MA. 85

Gibas, C. and P. Jambeck. 2002. **Introduction à la bioinformatique**. O'Reilly. 6

Gibson, D. G., J. I. Glass, C. Lartigue, V. N. Noskov, R. Y. Chuang, M. A. Algire, G. A. Benders, M. G. Montague, L. Ma, M. M. Moodie, C. Merryman, S. Vashee, R. Krishnakumar, N. Assad-Garcia, C. Andrews-Pfannkoch, E. A. Denisova, L. Young, Z. Q. Qi, T. H. Segall-Shapiro, C. H. Calvey, P. P. Parmar, r. Hutchison, C. A., H. O. Smith, and J. C. Venter. 2010. **Creation of a bacterial cell controlled by a chemically synthesized genome**. Science 329 :52–6. 126

Goldenfeld, N. and C. Woese. 2007. **Biology's next revolution**. Nature 445 :369. 106

Goloboff, P. A. 2005. **Minority-rule supertrees ? MRP, compatibility, and MinFlip may display the least frequent groups.** Cladistics 21 :282–294. 63

Goloboff, P. A. and D. Pol. 2002. **Semi-strict supertrees**. Cladistics 18 :514–525. 61, 63, 67

Goodman, M., J. Czelusniak, G. W. Moore, R. A. Herrera, and G. Matsuda. 1979. **Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences**. Syst. Zool. 28 :132–163. 106

Goodman, M., L. I. Grossman, and D. E. Wildman. 2005. **Moving primate genomics beyond the chimpanzee genome**. Trends Genet. 21 :511–517. 71

Gordon, A. G. 1986. **Consensus supertrees : the synthesis of rooted trees containing overlapping sets of labelled leaves**. J. Classif. 3 :335–348. 60, 61

Górecki, P. 2004. **Reconciliation problems for duplication, loss and horizontal gene transfer.** Pages 316–325 *in* RECOMB (P. E. Bourne and D. Gusfield, eds.) ACM. 108

Green, R. E., J. Krause, A. W. Briggs, T. Maricic, U. Stenzel, M. Kircher, N. Patterson, H. Li, W. Zhai, M. H. Fritz, N. F. Hansen, E. Y. Durand, A. S. Malaspinas, J. D. Jensen, T. Marques-Bonet, C. Alkan, K. Prufer, M. Meyer, H. A. Burbano, J. M. Good, R. Schultz, A. Aximu-Petri, A. Butthof, B. Hober, B. Hoffner, M. Siegemund, A. Weihmann, C. Nusbaum, E. S. Lander, C. Russ, N. Novod, J. Affourtit, M. Egholm, C. Verna, P. Rudan, D. Brajkovic, Z. Kucan, I. Gusic, V. B. Doronichev, L. V. Golovanova, C. Lalueza-Fox, M. de la Rasilla, J. Fortea, A. Rosas, R. W. Schmitz, P. L. Johnson, E. E. Eichler, D. Falush, E. Birney, J. C. Mullikin, M. Slatkin, R. Nielsen, J. Kelso, M. Lachmann, D. Reich, and S. Paabo. 2010. **A draft sequence of the neandertal genome**. Science 328 :710–22. 16

Grunewald, S., M. A. Steel, and M. S. Swenson. 2007. **Closure operations in phylogenetics**. Math Biosci 208 :521–537. 65, 67

Guan, X. and E. C. Uberbacher. 1996. **Alignments of DNA and protein sequences containing frameshift errors**. Comput. Appl. Biosci. 12 :31–40. 42

Guigo, R., I. Muchnik, and T. F. Smith. 1996. **Reconstruction of ancient molecular phylogeny**. Mol. Phylogenet. Evol. 6 :189–213. 106

Guillemot, S. and V. Berry. 2007. **Finding a largest subset of rooted triples identifying a tree is an NP-hard task**. Tech. rep. LIRMM, Univ. Montpellier 2. 74

Gusfield, D. 1991. **Efficient algorithms for inferring evolutionary trees**. Networks 21 :19–28. 96

Hallegger, M., M. Llorian, and C. W. Smith. 2010. **Alternative splicing : global insights**. Febs J 277 :856–66. 24

Hallett, M., J. Lagergren, and A. Tofigh. 2004. **Simultaneous identification of duplications and lateral transfers**. Pages 347–356 *in* RECOMB '04 ACM, New York, NY, USA. 106, 108, 110

Harel, D. and R. E. Tarjan. 1984. **Fast algorithms for finding nearest common ancestors**. SIAM J. Comput. 13 :338–355. 95

Hein, J. 1994. **An algorithm combining DNA and protein alignment**. J Theor Biol 167 :169–174. 41

Helfgott, D. and R. J. Mason-Gamer. 2004. **The evolution of North American Elymus (Triticeae, Poaceae) allotetraploids : evidence from phosphoenolpyruvate carboxylase gene sequences**. Systematic Botany 29 :850–861. 81

Henikoff, S. and J. G. Henikoff. 1992. **Amino acid substitution matrices from protein blocks**. Proc Natl Acad Sci U S A 89 :10915–9. 43

Higgins, D. G., A. J. Bleasby, and R. Fuchs. 1992. **CLUSTAL V : improved software for multiple sequence alignment**. Comput Appl Biosci 8 :189–191. 41, 50

Hillis, D. M. and J. P. Huelsenbeck. 1994. **Support for dental HIV transmission**. Nature 369 :24–5. 2

Hillis, D. M., J. P. Huelsenbeck, and C. W. Cunningham. 1994. **Application and accuracy of molecular phylogenies**. Science 264 :671–7. 2

Hofreiter, M. 2008. **Dna sequencing : Mammoth genomics**. Nature 456 :330–1. 16

Hogeweg, P. and B. Hesper. 1978. **Interactive instruction on population interactions**. Comput Biol Med 8 :319–27. 6

Huang, X. and A. Madan. 1999. **CAP3 : a DNA sequence assembly program**. Genome Res. 9 :868–877. 55

Hubbard, T. J., B. L. Aken, S. Ayling, B. Ballester, K. Beal, E. Bragin, S. Brent, Y. Chen, P. Clapham, L. Clarke, G. Coates, S. Fairley, S. Fitzgerald, J. Fernandez-Banet, L. Gordon, S. Graf, S. Haider, M. Hammond, R. Holland, K. Howe, A. Jenkinson, N. Johnson, A. Kahari, D. Keefe, S. Keenan, R. Kinsella, F. Kokocinski, E. Kulesha, D. Lawson, I. Longden, K. Megy, P. Meidl, B. Overduin, A. Parker, B. Pritchard, D. Rios, M. Schuster, G. Slater, D. Smedley, W. Spooner, G. Spudich, S. Trevanion, A. Vilella, J. Vogel, S. White, S. Wilder, A. Zadissa, E. Birney, F. Cunningham, V. Curwen, R. Durbin, X. M. Fernandez-Suarez, J. Herrero,

A. Kasprzyk, G. Proctor, J. Smith, S. Searle, and P. Flicek. 2009. **Ensembl 2009**. Nucleic Acids Res 37 :D690–7. 28

Huson, D., R. Rupp, and S. Celine. 2010. **Phylogenetic Networks : Concepts, Algorithms and Applications**. Cambridge University Press. 89

Huson, D. H., S. M. Nettles, and T. J. Warnow. 1999. **Disk-covering, a fast-converging method for phylogenetic tree reconstruction**. J Comput Biol 6 :369–86. 61

International Human Genome Sequencing Consortium. 2004. **Finishing the euchromatic sequence of the human genome**. Nature 431 :931–45. 14

Jacq, C., J. R. Miller, and G. G. Brownlee. 1977. **A pseudogene structure in 5S DNA of Xenopus laevis**. Cell 12 :109–20. 24

K. Yu. Gorbunov, V. A. L. 2010. **An algorithm of reconciliation of gene and species trees and inferring gene duplications, losses and horizontal transfers**. Information processes 10 :140–144. 106, 107, 109, 110

Katoh, K., K. Kuma, H. Toh, and T. Miyata. 2005. **MAFFT version 5 : improvement in accuracy of multiple sequence alignment**. Nucleic Acids Res 33 :511–518. 41

Kececioglu, J. and D. Starrett. 2004. **Aligning alignments exactly**. Pages 85–96 *in* RECOMB '04 : Proceedings of the eighth annual international conference on Resaerch in computational molecular biology ACM, San Diego, California, USA. 50

Kececioglu, J. and W. Zhang. 1998. **Aligning alignments**. Pages 189–208 *in* CPM '98 : Proceedings of the 9th Annual Symposium on Combinatorial Pattern Matching vol. 1448 of *Lecture Notes In Computer Science*. Springer-Verlag. 49

Keeling, P. J., G. Burger, D. G. Durnford, B. F. Lang, R. W. Lee, R. E. Pearlman, A. J. Roger, and M. W. Gray. 2005. **The tree of eukaryotes**. Trends Ecol Evol 20 :670–6. 104

Keese, P. 2008. **Risks from GMOs due to horizontal gene transfer**. Environ Biosafety Res 7 :123–49. 106

Kircher, M., U. Stenzel, and J. Kelso. 2009. **Improved base calling for the illumina genome analyzer using machine learning strategies**. Genome Biol. 10 :R83. 54

Kurland, C. G., B. Canback, and O. G. Berg. 2003. **Horizontal gene transfer : a critical view**. Proc. Natl. Acad. Sci. U.S.A. 100 :9658–9662. 106

Landan, G. and D. Graur. 2007. **Heads or tails : a simple reliability check for multiple sequence alignments**. Mol. Biol. Evol. 24 :1380–1383. 46

Lander, E. S., L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, R. Funke, D. Gage, K. Harris, A. Heaford, J. Howland, L. Kann, J. Lehoczky, R. LeVine, P. McEwan, K. McKernan, J. Meldrim, J. P. Mesirov, C. Miranda, W. Morris, J. Naylor, C. Raymond, M. Rosetti, R. Santos, A. Sheridan, C. Sougnez, N. Stange-Thomann, N. Stojanovic, A. Subramanian, D. Wyman, J. Rogers, J. Sulston, R. Ainscough, S. Beck, D. Bentley, J. Burton, C. Clee, N. Carter, A. Coulson, R. Deadman, P. Deloukas, A. Dunham, I. Dunham, R. Durbin, L. French, D. Grafham, S. Gregory, T. Hubbard, S. Humphray, A. Hunt, M. Jones, C. Lloyd, A. McMurray, L. Matthews, S. Mercer, S. Milne, J. C. Mullikin, A. Mungall, R. Plumb, M. Ross, R. Shownkeen, S. Sims, R. H. Waterston, R. K. Wilson, L. W. Hillier, J. D. McPherson, M. A. Marra, E. R. Mardis, L. A. Fulton, A. T. Chinwalla, K. H. Pepin, W. R. Gish, S. L. Chissoe, M. C. Wendl, K. D. Delehaunty, T. L. Miner, A. Delehaunty, J. B. Kramer, L. L. Cook, R. S. Fulton, D. L. Johnson, P. J. Minx, S. W. Clifton, T. Hawkins, E. Branscomb, P. Predki, P. Richardson, S. Wenning, T. Slezak, N. Doggett, J. F. Cheng, A. Olsen, S. Lucas, C. Elkin, E. Uberbacher, M. Frazier, et al. 2001. **Initial sequencing and analysis of the human genome**. Nature 409 :860–921. 13, 26

Levenshtein, V. 1966. **Binary codes capable of correcting deletions, insertions and reversals**. Soviet Physics Doklady 10 :707. 43

Li, L., J. Stoeckert, C. J., and D. S. Roos. 2003. **OrthoMCL : identification of ortholog groups for eukaryotic genomes**. Genome Res 13 :2178–89. 29

Li, M. K., L. Gu, S. S. Chen, J. Q. Dai, and S. H. Tao. 2008. **Evolution of the isochore structure in the scale of chromosome : insight from the mutation bias and fixation bias**. J Evol Biol 21 :173–82. 33

Libeskind-Hadas, R. and M. A. Charleston. 2009. **On the computational complexity of the reticulate cophylogeny reconstruction problem**. JCB 16 :105–117. 110

Linné, C. v. 1758. **Systema Naturæ**. 8

Loader, S., D. Pisani, J. Cotton, D. Gower, J. Day, and M. Wilkinson. 2007. **Relative time scales reveal multiple origins of parallel disjunct distributions of african caecilian amphibians**. Biol Lett. Pages 505,Äì–508. 109

Löytynoja, A. and N. Goldman. 2008. **Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis**. Science 320 :1632–1635. 41

Lynch, M. and J. S. Conery. 2000. **The evolutionary fate and consequences of duplicate genes**. Science 290 :1151–1155. 91

Margulies, M., M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben, J. Berka, M. S. Braverman, Y. J. Chen, Z. Chen, S. B. Dewell, L. Du, J. M.

Fierro, X. V. Gomes, B. C. Godwin, W. He, S. Helgesen, C. H. Ho, G. P. Irzyk, S. C. Jando, M. L. Alenquer, T. P. Jarvie, K. B. Jirage, J. B. Kim, J. R. Knight, J. R. Lanza, J. H. Leamon, S. M. Lefkowitz, M. Lei, J. Li, K. L. Lohman, H. Lu, V. B. Makhijani, K. E. McDade, M. P. McKenna, E. W. Myers, E. Nickerson, J. R. Nobile, R. Plant, B. P. Puc, M. T. Ronan, G. T. Roth, G. J. Sarkis, J. F. Simons, J. W. Simpson, M. Srinivasan, K. R. Tartaro, A. Tomasz, K. A. Vogt, G. A. Volkmer, S. H. Wang, Y. Wang, M. P. Weiner, P. Yu, R. F. Begley, and J. M. Rothberg. 2005. **Genome sequencing in microfabricated high-density picolitre reactors**. Nature 437 :376–380. 39, 54

Mason-Gamer, R. J. 2001. **Origin of North American Elymus (Poaceae : Triticeae) allotetraploids based on granule-bound starch synthase gene sequences**. Systematic Botany 26 :757–768. 81

Mason-Gamer, R. J. 2005. **The β-amylase genes of grasses and a phylogenetic analysis of the Triticeae (Poaceae)**. American Journal of Botany 92 :1045–1058. 81

Mason-Gamer, R. J., N. L. Orme, and C. M. Anderson. 2002. **Phylogenetic analysis of North American Elymus and the monogenomic Triticeae (Poaceae) using three chloroplast dna data sets**. Genome 45 :991–1002. 81

Mattick, J. S. 2003. **Challenging the dogma : the hidden layer of non-protein-coding RNAs in complex organisms**. Bioessays 25 :930–9. 22

McInerney, J. O., J. A. Cotton, and D. Pisani. 2008. **The prokaryotic Tree of Life : past, present... and future ?** Trends Ecol. Evol. (Amst.) 23 :276–281. 106

Meredith, R. W., J. Gatesy, W. J. Murphy, O. A. Ryder, and M. S. Springer. 2009. **Molecular decay of the tooth gene Enamelin (ENAM) mirrors the loss of enamel in the fossil record of placental mammals**. PLoS Genet 5 :e1000634. 24

Merkle, D. and M. Middendorf. 2005. **Reconstruction of the cophylogenetic history of related phylogenetic trees with divergence timing information.** Theory Biosci 123 :277–299. 109

Merkle, D., M. Middendorf, and N. Wieseke. 2010. **A parameter-adaptive dynamic programming approach for inferring cophylogenies**. BMC Bioinformatics 11 :S60. 109

Miller, W., D. I. Drautz, A. Ratan, B. Pusey, J. Qi, A. M. Lesk, L. P. Tomsho, M. D. Packard, F. Zhao, A. Sher, A. Tikhonov, B. Raney, N. Patterson, K. Lindblad-Toh, E. S. Lander, J. R. Knight, G. P. Irzyk, K. M. Fredrikson, T. T. Harkins, S. Sheridan, T. Pringle, and S. C. Schuster. 2008. **Sequencing the nuclear genome of the extinct woolly mammoth**. Nature 456 :387–90. 16

Mouchiroud, D., C. Gautier, and G. Bernardi. 1988. **The compositional distribution of coding sequences and DNA molecules in humans and murids**. J Mol Evol 27 :311–20. 33, 34

Moulton, V., C. Semple, and M. Steel. 2007. **Optimizing phylogenetic diversity under constraints**. J Theor Biol 246 :186–94. 2

Needleman, S. and C. Wunsch. 1970. **A general method applicable to the search for similarities in the amino acid sequence of two proteins**. J. Mol. Biol. 48 :443–453. 42, 44

Neumann, D. A. 1983. **Faithful consensus methods for n-trees**. Matematical Biosciences 63 :271–287. 68

Newberg, L. A. 2008. **Memory-efficient dynamic programming backtrace and pairwise local sequence alignment**. Bioinformatics 24 :1772–8. 57

Notredame, C., D. G. Higgins, and J. Heringa. 2000. **T-Coffee : a novel method for fast and accurate multiple sequence alignment**. J Mol Biol 302 :205–217. 41

Ochiai, K., T. Yamanaka, K. Kimura, and O. Sawada. 1959. **Inheritance of drug resistance (and its transfer) between Shigella strains and between Shigella and E. coli strains (in japanese)**. Hihon Iji Shimpor 1861 :34. 106

Ohno, S. 1970. **Evolution by Gene duplication** vol. 24. Springer-Verlag, New York. 91

Orgel, L. E., F. H. Crick, and C. Sapienza. 1980. **Selfish DNA**. Nature 288 :645–6. 26

Page, R. D. 1994. **Maps between trees and cladistic analysis of historical associations among genes, organisms, and areas**. Syst. Biol. 43 :58–77. 106

Page, R. D. 2007. **TBMap : a taxonomic perspective on the phylogenetic database TreeBASE**. BMC Bioinformatics 8 :158. 85

Pedersen, C. N. S., R. B. Lyngsø, and J. Hein. 1998. **Comparison of coding DNA**. Pages 153–173 *in* CPM 98 : combinatorial pattern matching (S. B. . Heidelberg, ed.) vol. 1448 of *Lecture Notes in Computer Science*. 41

Penel, S., A. M. Arigon, J. F. Dufayard, A. S. Sertier, V. Daubin, L. Duret, M. Gouy, and G. Perrière. 2009. **Databases of homologous gene families for comparative genomics**. BMC Bioinformatics 10 Suppl 6 :S3. 20, 89, 102

Pesole, G., G. Grillo, A. Larizza, and S. Liuni. 2000. **The untranslated regions of eukaryotic mRNAs : structure, function, evolution and bioinformatic tools for their analysis**. Brief Bioinform 1 :236–49. 22

Petersen, G. and O. Seberg. 1997. **Phylogenetic analysis of the triticeae (poaceae) based on rpoa sequence data**. Molecular Phylogenetics and Evolution 7 :217–230. 81

Petersen, G. and O. Seberg. 2002. **Molecular evolution and phylogenetic application of DMC1**. Molecular Phylogenetics and Evolution 22 :43–50. 81

Philippe, H., N. Lartillot, and H. Brinkmann. 2005. **Multigene analyses of bilaterian animals corroborate the monophyly of Ecdysozoa, Lophotrochozoa, and Protostomia**. Mol Biol Evol 22 :1246–53. 104

Pisani, D. and M. Wilkinson. 2002. **Matrix representation with parsimony, taxonomic congruence, and total evidence**. Syst. Biol. 51 :151–155. 63

Pollard, D. A., V. N. Iyer, A. M. Moses, and M. B. Eisen. 2006. **Widespread discordance of gene trees with species tree in Drosophila : evidence for incomplete lineage sorting**. PLoS Genetics 2 :1634–1647. 84

Ponting, C. P., J. Schultz, F. Milpetz, and P. Bork. 1999. **SMART : identification and annotation of domains from signalling and extracellular protein sequences**. Nucleic Acids Res. 27 :229–232. 51

Posada, D. 2003. **Using MODELTEST and PAUP* to select a model of nucleotide substitution**. Curr Protoc Bioinformatics Chapter 6 :Unit 6 5. 31

Puigbo, P., Y. Wolf, and E. Koonin. 2009. **Search for a 'Tree of Life' in the thicket of the phylogenetic forest**. Journal of Biology 8 :59. 106

Purvis, A. 1995a. **A composite estimate of primate phylogeny**. Philos Trans R Soc Lond B Biol Sci 348 :405–21. 60

Purvis, A. 1995b. **A modification to Baum and Ragan's method for combining phylogenetic trees**. Syst. Biol. 44 :251–255. 63

Ragan, M. A. 1992. **Matrix representation in reconstructing phylogenetic relationships among the eukaryots**. Biosystems 28 :47–55. 61

Raghava, G. P., S. M. Searle, P. C. Audley, J. D. Barber, and G. J. Barton. 2003. **OXBench : a benchmark for evaluation of protein multiple sequence alignment accuracy**. BMC Bioinformatics 4 :47. 51

Rambaut, A. 2002. **PhyloGen : phylogenetic tree simulator package**. 116

Ranwez, V. 2002. **Méthodes efficaces pour reconstruire de grandes phylogénies suivant le principe du maximum de vraisemblance**. Ph.D. thesis Montpellier II. 31

Ranwez, V., V. Berry, A. Criscuolo, P. H. Fabre, S. Guillemot, C. Scornavacca, and E. J. Douzery. 2007a. **PhySIC : a veto supertree method with desirable properties**. Syst Biol 56 :798–817. 59, 64, 67, 68, 69, 74

Ranwez, V., N. Clairon, F. Delsuc, S. Pourali, N. Auberval, S. Diser, and V. Berry. 2009. **PhyloExplorer : a web server to validate, explore and query phylogenetic trees**. BMC Evol Biol 9 :108. 60

Ranwez, V., A. Criscuolo, and E. J. Douzery. 2010. **SuperTriplets : a triplet-based supertree approach to phylogenomics**. Bioinformatics 26 :i115–23. 63

Ranwez, V., F. Delsuc, S. Ranwez, K. Belkhir, M. K. Tilak, and E. J. Douzery. 2007b. **OrthoMaM : a database of orthologous genomic markers for placental mammal phylogenetics**. BMC Evol Biol 7 :241. 21, 29

Ranwez, V., S. Harispe, F. Delsuc, and E. J. Douzery. soumis. **MACSE : Multiple Alignment of Coding SEquences**. submitted to Mol Biol Evol . 22, 39

Ray, J. 1686. **Historia plantarum**. 8

Reenskaug, T. 1979. **Models - Views - Controllers**. Tech. rep. Technical Note, Xerox Parc. 85

Robinson, M., C. Gautier, and D. Mouchiroud. 1997. **Evolution of isochores in rodents**. Mol Biol Evol 14 :823–8. 33

Rodriguez-Ezpeleta, N., H. Brinkmann, B. Roure, N. Lartillot, B. F. Lang, and H. Philippe. 2007. **Detecting and overcoming systematic errors in genome-scale phylogenies**. Syst Biol 56 :389–99. 104

Rokas, A. and S. B. Carroll. 2006. **Bushes in the Tree of Life**. PLoS Biol 4 :e352. 16

Romiguier, J., V. Ranwez, E. J. Douzery, and N. Galtier. 2010. **Contrasting GC-content dynamics across 33 mammalian genomes : Relationship with life-history traits and chromosome sizes**. Genome Res 20 :1001–9. 21, 33, 34

Russell, R. D. and A. T. Beckenbach. 2008. **Recoding of translation in turtle mitochondrial genomes : programmed frameshift mutations and evidence of a modified genetic code**. J. Mol. Evol. 67 :682–695. 39

Sanderson, M. J., M. J. Donoghue, W. Piel, and T. Eriksson. 1994. **TreeBASE : a prototype database of phylogenetic analyses and an interactive tool for browsing the phylogeny of life**. Amer. Jour. Bot. 81 :183. 84

Sankoff, D. 1972. **Matching sequences under deletion-insertion constraints**. Proc Natl Acad Sci U S A 69 :4–6. 44

Scornavacca, C. 2009. **Supertree methods for phylogenomics**. Ph.D. thesis Montpellier II. 60, 75, 77, 95

Scornavacca, C., V. Berry, V. Lefort, E. J. P. Douzery, and V. Ranwez. 2008. **PhySIC_IST : cleaning source trees to infer more informative supertrees**. BMC Bioinformatics 9 :413. 59, 75, 81, 89

Scornavacca, C., V. Berry, and V. Ranwez. 2009. **From gene trees to species trees through a supertree approach**. Pages 702–714 *in* LATA '09 : Proceedings of the 3rd International Conference on Language and Automata Theory and Applications vol. 5457 of *LNCS* Springer-Verlag, Berlin, Heidelberg. 92, 93

Scornavacca, C., V. Berry, and V. Ranwez. sous presse. **Building species trees from larger parts of phylogenomic databases**. Information and Computation . 92, 93, 95, 97, 98, 103

Sellers, P. 1974. **On the theory and computation of evolutionary distances**. SIAM J. Appl. Match Pages 787–793. 43

Semple, C. and M. Steel. 2000. **A supertree method for rooted trees**. Discrete Appl. Math. 105 :147–158. 62

Semple, C. and M. Steel. 2003. **Phylogenetics (Oxford Lecture Series in Mathematics and Its Applications, 24)**. Oxford University Press, USA. 64, 72

Spencer, C. C., P. Deloukas, S. Hunt, J. Mullikin, S. Myers, B. Silverman, P. Donnelly, D. Bentley, and G. McVean. 2006. **The influence of recombination on human genetic diversity**. PLoS Genet 2 :e148. 26

Steel, M., A. Dress, and S. Böcker. 2000. **Simple but fundamental limitations on supertree and consensus tree methods**. Syst Biol 49 :363–368. 62, 68

Steel, M. A. 1992. **The complexity of reconstructing trees from qualitative characters and subtree**. J. Classif. 9 :91–116. 100

Stocsits, R. R., I. L. Hofacker, C. Fried, and P. F. Stadler. 2005. **Multiple sequence alignments of partially coding nucleic acid sequences**. BMC Bioinformatics 6 :160. 41, 42

Suyama, M., D. Torrents, and P. Bork. 2006. **PAL2NAL : robust conversion of protein sequence alignments into the corresponding codon alignments**. Nucleic Acids Res. 34 :W609–612. 38

Swami, M. 2010. **Small RNAs : Pseudogenes act as microRNA decoys**. Nat Rev Cancer 10 :535. 24

Swofford, D. L. 2003. **PAUP* : Phylogenetic Analysis Using Parsimony (*and Other Methods). Version 4.** Sinauer Associates, Sunderland, Massachusetts. 82

Thompson, J. D., P. Koehl, R. Ripp, and O. Poch. 2005. **BAliBASE 3.0 : latest developments of the multiple sequence alignment benchmark**. Proteins 61 :127–36. 51

Thorley, J., M. Wilkinson, and M. Charleston. 1998. **The information content of consensus trees**. Pages 91–98 *in* Advances in Data Science and Classification. Studies in Classification, Data Analysis, and Knowledge Organization (A. Rizzi, M. Vichi, and H.-H. Bock, eds.). 72

Thorley, J. L. and M. Wilkinson. 2003. **A view of supertrees methods**. Pages 185–194 *in* Bioconsensus (M. F. Janowitz, F.-J. Lapointe, F. R. McMorris, and F. S. Roberts, eds.) vol. 61 of *Discrete Mathematics and Theoretical Computer Science* DIMACS. 61

Tofigh, A., M. Hallett, and J. Lagergren. 2010. **Simultaneous identification of duplications and lateral gene transfers**. IEEE/ACM TCBB 99. 108, 109, 110

Tofigh, A., J. Sjöstrand, B. Sennblad, L. Arvestad, and J. Lagergren. communication personnelle. **Detecting LGTs using a novel probabilistic model integrating duplications, LGTs, losses, rate variation, and sequence evolution**. 106, 110

Venter, J. C., M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, J. D. Gocayne, P. Amanatides, R. M. Ballew, D. H. Huson, J. R. Wortman, Q. Zhang, C. D. Kodira, X. H. Zheng, L. Chen, M. Skupski, G. Subramanian, P. D. Thomas, J. Zhang, G. L. Gabor Miklos, C. Nelson, S. Broder, A. G. Clark, J. Nadeau, V. A. McKusick, N. Zinder, A. J. Levine, R. J. Roberts, M. Simon, C. Slayman, M. Hunkapiller, R. Bolanos, A. Delcher, I. Dew, D. Fasulo, M. Flanigan, L. Florea, A. Halpern, S. Hannenhalli, S. Kravitz, S. Levy, C. Mobarry, K. Reinert, K. Remington, J. Abu-Threideh, E. Beasley, K. Biddick, V. Bonazzi, R. Brandon, M. Cargill, I. Chandramouliswaran, R. Charlab, K. Chaturvedi, Z. Deng, V. Di Francesco, P. Dunn, K. Eilbeck, C. Evangelista, A. E. Gabrielian, W. Gan, W. Ge, F. Gong, Z. Gu, P. Guan, T. J. Heiman, M. E. Higgins, R. R. Ji, Z. Ke, K. A. Ketchum, Z. Lai, Y. Lei, Z. Li, J. Li, Y. Liang, X. Lin, F. Lu, G. V. Merkulov, N. Milshina, H. M. Moore, A. K. Naik, V. A. Narayan, B. Neelam, D. Nusskern, D. B. Rusch, S. Salzberg, W. Shao, B. Shue, J. Sun, Z. Wang, A. Wang, X. Wang, J. Wang, M. Wei, R. Wides, C. Xiao, C. Yan, et al. 2001. **The sequence of the human genome**. Science 291 :1304–51. 13

Vernot, B., M. Stolzer, A. Goldman, and D. Durand. 2008. **Reconciliation with non-binary species trees**. J. Comput. Biol. 15 :981–1006. 106, 108

Wang, L. and T. Jiang. 1994. **On the complexity of multiple sequence alignment**. J Comput Biol 1 :337–48. 49

Watson, J. D. and F. H. Crick. 1953. **Genetical implications of the structure of deoxyribonucleic acid**. Nature 171 :964–7. 11, 12

Webster, M. T. and N. G. Smith. 2004. **Fixation biases affecting human SNPs**. Trends Genet 20 :122–6. 26, 37

Wheeler, T. J. and J. D. Kececioglu. 2007. **Multiple alignment by aligning alignments**. Bioinformatics 23 :i559–i568. 50, 57

Wilkinson, M., J. A. Cotton, C. Creevey, O. Eulenstein, S. R. Harris, F. J. Lapointe, C. Levasseur, J. O. McInerney, D. Pisani, and J. L. Thorley. 2005. **The shape of supertrees to come : Tree shape related properties of fourteen supertree methods**. Systematic Biology 54 :419–431. 63

Wilkinson, M., J. A. Cotton, F. J. Lapointe, and D. Pisani. 2007. **Properties of supertree methods in the consensus setting**. Syst Biol 56 :330–7. 62

Wilkinson, M., J. Thorley, D. Pisani, F.-J. Lapointe, and O. McInerney. 2004. **Some desiderata for liberal supertree**. Pages 227–246 *in* Phylogenetic supertrees (combining information to reveal the Tree of Life) (O. R. P. Bininda-Emonds, ed.) vol. 4. Kluwer academic publishers. 68

Wilkinson, M., J. L. Thorley, D. Littlewood, and R. Bray. 2001. **Towards a phylogenetic supertree for Patyhelminthes ?** Pages 292–301 *in* Interrelationships of the Platyhelminthes (D. T. L. Littlewood and R. A. Bray, eds.). Taylor & Francis, London. 63

Williams, I., J. Richardson, A. Starkey, and I. Stansfield. 2004. **Genome-wide prediction of stop codon readthrough during translation in the yeast saccharomyces cerevisiae**. Nucleic Acids Res 32 :6605–16. 39

Winckler, W., S. R. Myers, D. J. Richter, R. C. Onofrio, G. J. McDonald, R. E. Bontrop, G. A. McVean, S. B. Gabriel, D. Reich, P. Donnelly, and D. Altshuler. 2005. **Comparison of fine-scale recombination rates in humans and chimpanzees**. Science 308 :107–11. 37

Wojtowicz, W. M., J. J. Flanagan, S. S. Millard, S. L. Zipursky, and J. C. Clemens. 2004. **Alternative splicing of Drosophila Dscam generates axon guidance receptors that exhibit isoform-specific homophilic binding**. Cell 118 :619–33. 24

Yamane, K. and T. Kawahara. 2005. **Intra- and interspecific phylogenetic relationships among diploid Triticum-Aegilops species (Poaceae) based on base-pair substitutions, indels, and microsatellites in chloroplast noncoding sequences**. American Journal of Botany 92 :1887–1898. 81

Yang, Z. and B. Rannala. 1997. **Bayesian phylogenetic inference using DNA sequences : a Markov Chain Monte Carlo Method**. Mol Biol Evol 14 :717–24. 116

Zhang, J. 2003. **Evolution by gene duplication : an update**. Trends in Ecology and Evolution 18 :292—298. 91

Zhang, L. 1997. **On a Mirkin-Muchnik-Smith conjecture for comparing molecular phylogenies**. Journal of Computational Biology 4 :177–187. 108

Zou, X. H., F.-M. Zhang, J.-G. Zhang, L.-L. Zang, L. Tang, J. Wang, T. Sang, and S. Ge. 2008. **Analysis of 142 genes resolves the rapid diversification of the rice genus**. Genome Biology 9 :R49. 84

Résumé

La phylogénie moléculaire est un champ de recherche qui étudie l'histoire évolutive d'espèces à partir de séquences moléculaires issues de leurs génomes. Les progrès technologiques nous permettent désormais d'avoir facilement accès aux génomes complets d'un nombre toujours plus grand d'espèces. On parle maintenant de "phylogénomique" pour désigner les études phylogénétiques qui se basent sur l'analyse de génomes complets. Parallèlement, le nombre d'espèces étudiées augmente lui aussi très rapidement, le but en ligne de mire (même si on le sait inatteignable) étant de proposer la phylogénie de l'ensemble des êtres vivants : "l'Arbre de la Vie". Cet objectif est avant tout un défi intellectuel qui reflète l'envie intarissable de l'homme de connaître et de comprendre le monde qui l'entoure. Mais cette connaissance, toujours meilleure, de l'Arbre de la Vie est également un outil clef pour mieux protéger ce monde fragile. En effet, la préservation des espèces est un enjeu majeur de notre siècle. La phylogénie moléculaire fournit un cadre théorique permettant une définition formelle de la notion, souvent floue, de biodiversité. L'obtention d'un Arbre de la Vie (même partiel) est donc un outil précieux qui permet de définir de manière objective les espèces (ou les zones) à protéger prioritairement.

L'obtention d'alignements de séquences orthologues est l'une des premières étapes de nombreuses analyses moléculaires et phylogénétiques. La qualité de cette étape est primordiale et conditionne, en partie, la validité des résultats obtenus ultérieurement sur la base de ces données. Ceci explique que nous ayons mené un effort particulier pour constituer la base de données OrthoMaM qui propose un ensemble de marqueurs phylogénétiques conservés en simple copie à l'échelle des mammifères. Nous avons également proposé un algorithme, basé sur une approche de programmation dynamique, qui permet d'aligner des séquences nucléotidiques codantes, en prenant en compte à la fois leurs traductions en acides aminés et la possibilité de changements de cadre de lecture. Sur la base d'un alignement des séquences orthologues d'un gène, on peut inférer son histoire évolutive. On utilise généralement plusieurs de ces histoires pour inférer l'arbre des espèces.

Les méthodes de super-arbre permettent de combiner des phylogénies partielles (aussi appelés arbres sources) en une seule phylogénie plus large offrant une synthèse des différents arbres sources. L'une des difficultés inhérentes aux méthodes de super-arbre est l'utilisation d'arbres sources incongruents, c'est-à-dire en désaccord sur la position phylogénétique de certaines espèces. Parce qu'elles permettent de combiner des arbres en une phylogénie toujours plus grande, les méthodes de super-arbre font partie des outils clefs pour assembler l'Arbre de la Vie. Nous avons formalisé deux propriétés qu'elles devraient respecter dans ce contexte. La première assure que toute information du super-arbre est présente (ou induite) par les arbres sources ; tandis que la seconde assure que le super-arbre ne contredit aucune des informations présentes (ou induites) par les arbres sources. Après avoir montré l'intérêt de ces propriétés en les comparant à d'autres, nous avons proposé une méthode de super-arbre, *PhySIC_IST*, qui les respecte. De plus, en introduisant un pré-traitement statistique (STC) qui élimine les résolutions minoritaires des arbres sources, nous avons généré toute une graduation de méthodes de super-arbre (STC + *PhySIC_IST*) qui, selon le paramètre STC utilisé, sont plus ou moins stringeantes sur les critères requis pour qu'une information soit présente dans le super-arbre.

Au cours de l'évolution, il est fréquent que des gènes soient dupliqués ou perdus au sein d'un génome. Il arrive également qu'il y ait un transfert de patrimoine génétique entre deux espèces contemporaines vivant dans un même environnement. Ces évènements engendrent des différences entre l'histoire des espèces (ou arbre des espèces) et celle d'un gène les ayant subits (ou arbre de gènes). De plus, une espèce peut alors étiqueter plusieurs feuilles d'un arbre de gènes ce qui le rend inutilisable par les approches classiques de super-arbres. Nos travaux permettent d'intégrer le signal phylogénétique de tels arbres dans une analyse de type super-arbre visant à inférer l'Arbre de la Vie. Ce travail est l'un des premiers à permettre d'exploiter cette source d'information, qui est pourtant essentielle. Parallèlement nous avons développé un algorithme polynomial efficace pour inférer le scénario le plus parcimonieux (en terme de duplication, perte et transfert) permettant d'expliquer les différences entre un arbre de gènes et un arbre d'espèces.

Les travaux décrits dans ce manuscrit vont de l'informatique théorique à l'étude de données biologiques en passant par le développement de logiciels et de services Web. Cependant, ils s'inscrivent tous dans une perspective à long terme qui vise à obtenir, à partir des génomes actuels, un Arbre de la Vie fiable, complet et annoté.

Mots-clefs : Phylogénie, Phylogénomique, Arbre de la Vie, Algorithmes, Programmation dynamique, Super-arbre, Alignement de séquences, Familles multigéniques.