Devoir 5. Comparaison d'arbres non-ordonnés

5.0 Arbres non-ordonnés et phylogénies

Afin de représenter un arbre où l'ordre des enfants n'est pas important, la structure la plus convenable est un arbre ordonné. Le but de cet exercice est de développer un cadre algorithmique pour la comparaison d'arbres enracinés (non-ordonnés). Le défi est de détecter des arbres équivalents avec l'ordonnance arbitraire[1] imposée par l'implémentation (Fig. 1).

Une application des arbres non-ordonnés est en biologie : un arbre évolutif, ou **phylogénie** d'espèces est un arbre enraciné où les nœuds externes correspondent aux espèces et les nœuds internes ont degré > 1 (Fig. 2). Tout nœud interne x correspond à un ancêtre hypothétique pour un sous-ensemble d'organismes existants. Notez que dans ce contexte, les nœuds externes sont nommés ou «étiquetés», et donc ils ont des identités distinctes. On définit le **clade** $C(x)$ pour chaque nœud x comme l'ensemble de nœuds externes dans son sous-arbre :

$$C(x) = \begin{cases} \{x\} & \text{si } x \text{ est externe ;} \\ \bigcup_{y \in \mathsf{enfants}(x)} C(y) & \text{si } x \text{ est interne} \end{cases}$$

Clairement, la collection de ses clades $\mathcal{C}(T) = \{C(x) : x \in T\}$ détermine[2] sans ambiguïté l'arbre T.

[1] Normalement, on met les enfants d'un nœud x soit dans un tableau $x.\mathsf{children}[0..d-1]$, soit dans deux variables $x.\mathsf{left}$ et $x.\mathsf{right}$ (si arbre binaire).

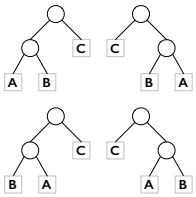

FIG. 1: Quatre arbres enracinés qui correspondent au même arbre non-ordonné.

[2] On peut reconstruire T à partir de $\mathcal{C}$: créer un nœud pour tout $C \in \mathcal{C}(T)$, et les lier entre eux — C est la descendante de C' si $C \subset C'$.

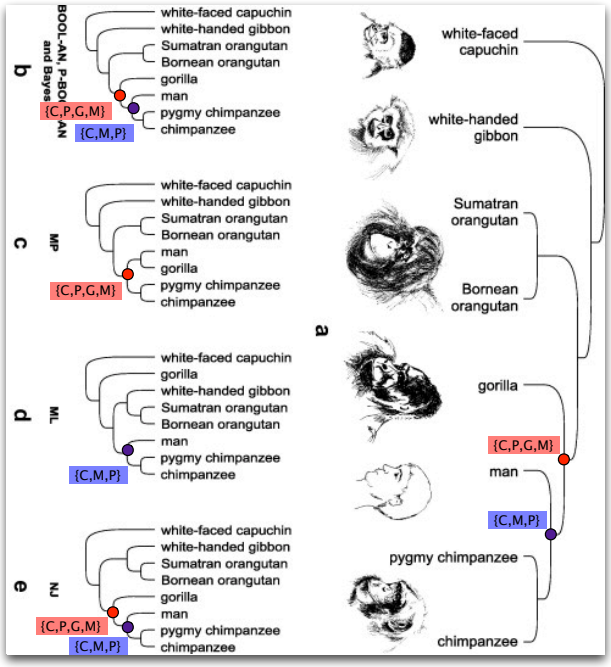

FIG. 2: Phylogénies de primates. Est-ce que les arbres montrent les mêmes relations entre les primates ? Non, parce que sur **c** on n'a pas d'ancêtre commun exclusif aux chimpanzés et l'humain (clade $\{\mathsf{C},\mathsf{M},\mathsf{P}\}$). Par contre, **c** contient un ancêtre commun aux chimpanzés, l'humain et le gorille (clade $\{\mathsf{C},\mathsf{P},\mathsf{G},\mathsf{M}\}$) mais **d** n'a pas de tel nœud interne. [Ari & al 2012 DOI: 10.1016/j.ympev.2012.01.010]

5.1 Structure de donnnées pour les clades

En général, on caractérise la différence entre deux arbres par les clades qui sont exclusifs à l'un ou l'autre. On veut donc développer une structure qui permet la comparaison rapide de clades entre deux arbres. Supposons que les nœuds externes sont étiquetés par $0, 1, 2, \ldots, n-1$ et que tout nœud interne a au moins deux enfants. Chaque clade est alors encodé comme un sous-ensemble $C(x) \subseteq \{0, 1, 2, \ldots, n-1\}$ des étiquettes. Si on veut comparer deux arbres, il faut qu'on travaille avec une structure de données qui permet la recherche entre tels sous-ensembles de nombres entiers. L'idée de clé est de numéroter les nœuds externes dans le premier arbre dans un parcours post-fixe, de gauche à droite. En conséquence, un clade correspond à une rangée d'indices $L(x)..R(x)$, où $L(x)$ est le minimum et $R(x)$ est le maximum des indices dans le sous-arbre de x, ce qu'on peut facilement calculer (Fig. 3).

Indexage de nœuds externes : D'abord, on parcourt l'arbre pour déterminer l'ordre des nœuds externes de gauche à droite, en remplissant[3] un tableau $\mathsf{idx}[0..n-1]$. La cellule $\mathsf{idx}[x]$ contient l'**indice** du nœud externe x ; le premier nœud visité obtient l'indice 0, le deuxième obtient 1, etc.

Calcul de clades : Par construction, chaque clade $C(x)$ contient des nœuds avec indices consécutives dans l'arbre. En particulier, si $C(x) = \{y_1, y_2, \ldots, y_k\}$, alors $\{\mathsf{idx}[y_1], \mathsf{idx}[y_2], \ldots, \mathsf{idx}[y_k]\} = \{L(x), L(x)+1, \ldots, R(x)\}$ avec $L(x) = \min_i \mathsf{idx}[y_i]$ et $R(x) = \max_i \mathsf{idx}[y_i]$. Soit $N(x)$ le nombre de nœuds externes dans le sous-arbre de x. On a alors les récurrences suivantes :

$$L(x) = \begin{cases} \mathsf{idx}[x] & \text{si } x \text{ est externe ;} \\ \min_{y \in \mathsf{enfants}(x)} \{L(y)\} & \text{si } x \text{ ext interne} \end{cases} \tag{5.1a}$$

$$R(x) = \begin{cases} \mathsf{idx}[x] & \text{si } x \text{ est externe ;} \\ \max_{y \in \mathsf{enfants}(x)} \{R(y)\} & \text{si } x \text{ ext interne} \end{cases} \tag{5.1b}$$

$$N(x) = \begin{cases} 1 & \text{si } x \text{ est externe ;} \\ \sum_{y \in \mathsf{enfants}(x)} N(y) & \text{si } x \text{ ext interne} \end{cases} \tag{5.1c}$$

Dans les exercices suivants, assumez un arbre binaire où $x.\mathsf{left}$ et $x.\mathsf{right}$ donnent les enfants d'un nœud interne x.

a. ▶ Donnez un algorithme récursif[4] qui remplit le tableau idx.

b. ▶ Donnez un algorithme récursif[5] qui calcule $L(x), R(x)$ (en utilisant $\mathsf{idx}[]$), ainsi que $N(x)$ à chaque nœud x par les récurrences de (5.1).

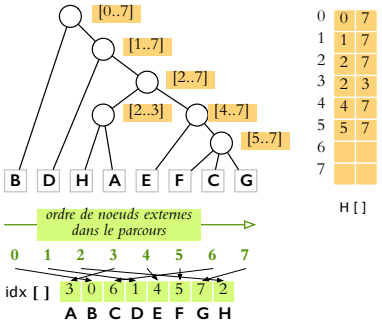

FIG. 3: Indexage et calcul de clades sur l'arbre. À partir de l'indexage dans $\mathsf{idx}[]$, on calcule les rangées $L..R$ indiquées aux nœuds internes. Le tableau H stocke le clade $L..R$ soit dans cellule L soit dans cellule R.

[3] En pratique, on se sert d'un tableau de hachage entre le nom du nœud externe et son indice.

[4] ***Indice:*** Faire un parcours postfixe : passer l'indice courant comme argument, et retourner le nombre de nœuds externes dans le sous-arbre.

[5] ***Indice:*** retourner le triple (L, R, N) lors d'un parcours postfixe

5.2 Comparaison de deux arbres

Maintenant, on peut comparer deux arbres T_1, T_2 sur le même ensemble de nœuds externes. D'abord, il faut déterminer et stocker idx et les rangées $L..R$ dans un parcours postfixe du premier arbre T_1. Ensuite, on calcule[6] L, R, N à chaque nœud dans le deuxième arbre T_2 selon les récurrences de (5.1), mais toujours avec les indices idx *du premier arbre* T_1. À chaque nœud interne $x' \in T_2$, après avoir déterminé $L(x')$, $R(x')$ et $N(x')$, on fait le test suivant :

(i) si $N(x') \neq R(x') - L(x') + 1$, alors x' est absent de T_1 ;

(ii) si $N(x') = R(x') - L(x') + 1$, alors x' est présent dans T_1 si et seulement si $L(x')..R(x')$ est l'intervalle d'indices pour un clade quelconque dans T_1.

Pour stocker les intervalles de T_1, on remplit un tableau $H[0..n-1]$ pendant le parcours de T_1. Dès qu'on détermine l'intervalle $L..R = L(x)..R(x)$ au nœud interne $x \in T_1$, on met $L..R$ soit dans la cellule $H[R]$, soit dans la cellule $H[L]$. (Si x est le premier enfant (gauche) de son parent, on place $H[R] \leftarrow L..R$; sinon $H[L] \leftarrow L..R$.) Ainsi le test en (ii) s'effectue en $\Theta(1)$ — il suffit d'examiner $H[L(x')]$ et $H[R(x')]$.

En tout, on performe la comparaison de deux arbres en $\Theta(n)$ temps (deux parcours), et avec un espace de travail $\Theta(n)$ (tableaux idx et H) !

▶ Montrez que le remplissage de H est correct : quand on enregistre le clade pour nœud $x \in T_1$, soit $H[R(x)]$ est vide (quand x est le premier enfant), soit $H[L(x)]$ est vide (si x n'est pas le premier enfant).

Indice: Suivez le parcours avec quelques exemples sur papier pour voir pourquoi cela est vrai.

[6] Dans l'exemple des phylogénies, le premier parcoues établit l'indexage des espèces : chimpanzé $\mapsto 0$, bonobo $\mapsto 1$, humain $\mapsto 2$, gorille $\mapsto 3, \ldots$. On utilise le même indexage dans les autres arbres (où un clade ne contient plus nécéssairement des indices consécutifs).

Remise de travail

Soumettez votre travail en forme d'un fichier PDF (algorithmes de §5.1a-b, preuve de §5.2) en Studium (date limite : 1er décembre, période de grâce jusqu'au 6e).