Apport des approches phylogénétiques pour expliquer l'origine des génomes mosaïques, exemple chez le Riz

Charles-Elie Rabier

Vincent Berry, Manuel Labous, Fabio Pardi et Céline Scornavacca

ISEM, Institut des Sciences de l'Evolution de Montpellier
LIRMM, Laboratoire d'informatique, de Robotique et de Microélectronique
Genome Harvest

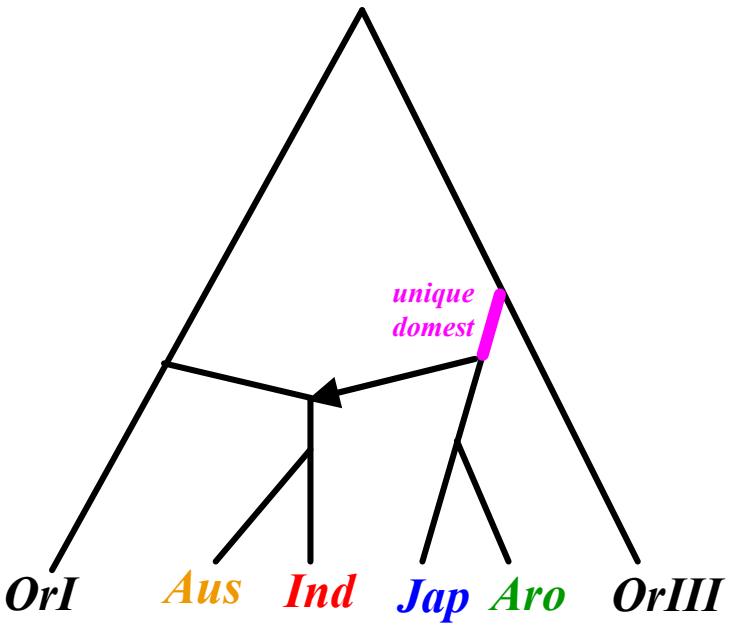
Collaborations : Jean-Christophe Glaszmann (CIRAD), Joao Santos (CIRAD)

Plan

Plan

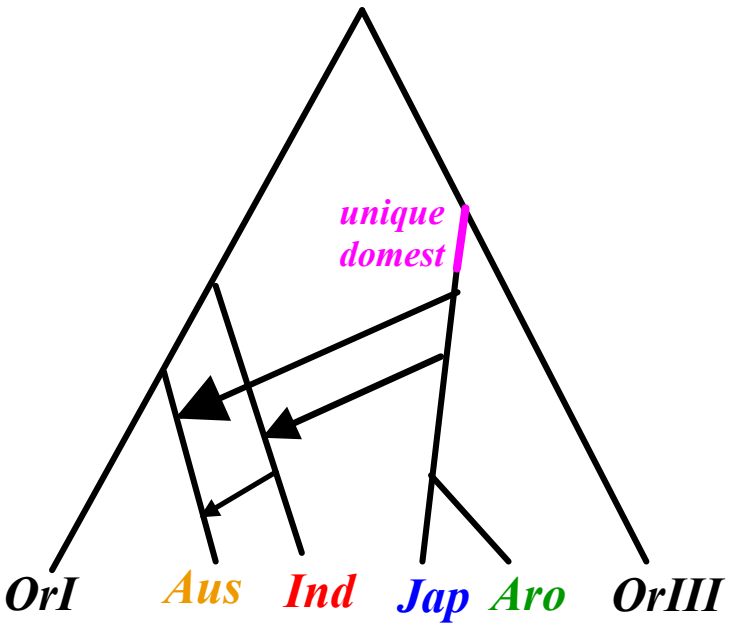
Quelques thèses sur la domestication

- Huang et al. (Nature, 2012) : japonica domestiqué à partir d'un riz sauvage dans le sud de la Chine, puis croisé à un sauvage dans le sud est de l'Asie, générant indica



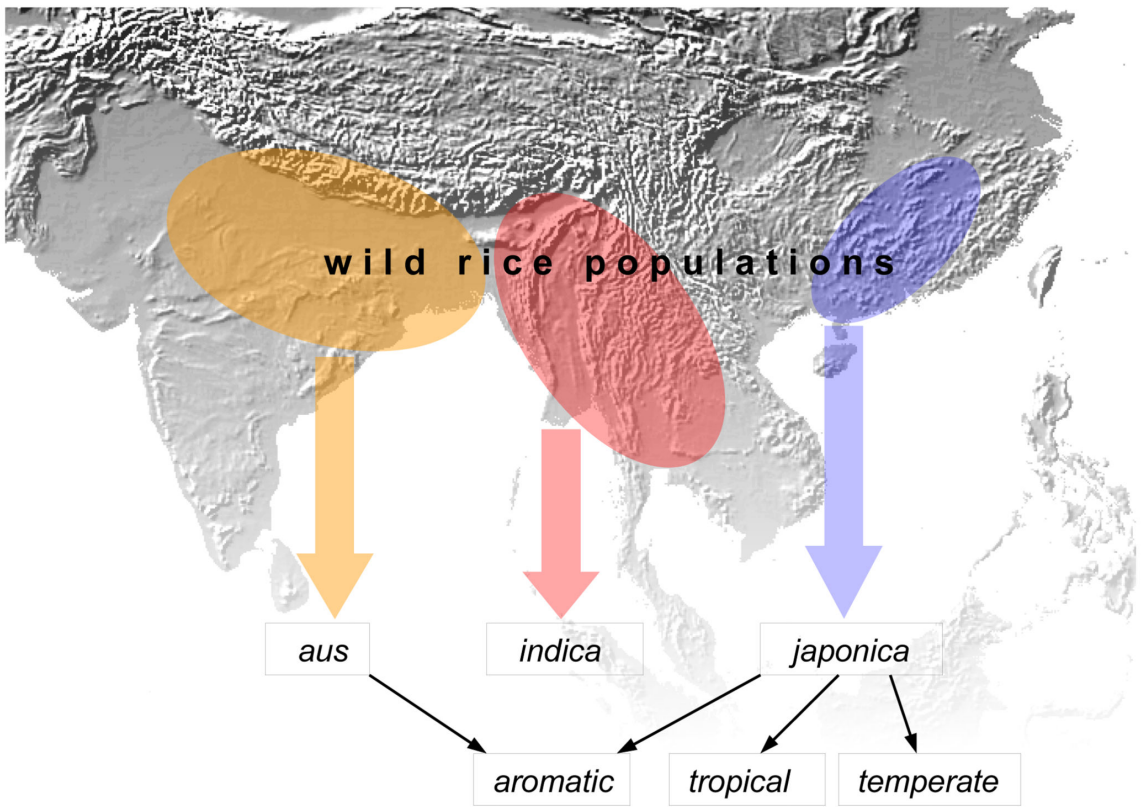
Quelques thèses sur la domestication

- Choi et al. (MBE, 2017) soutiennent aussi un seul évènement de domestication (japonica). Introgression par hybridation de japonica et proto-indica et proto-aus, générant indica et aus



Quelques thèses sur la domestication

- Civan et al. (Nature Plants, 2015) : indica, japonica et aus domestiqués séparément dans différentes parties d'Asie



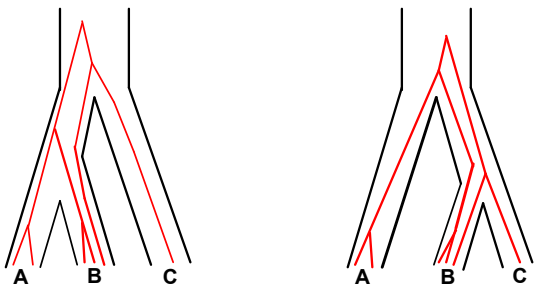
Notre approche méthodologique

On s'intéresse à un modèle qui, outre le tri de lignées, considère explicitement les mutations et hybridation. Modélisation Bayésienne plus fine.

Nos pistes :

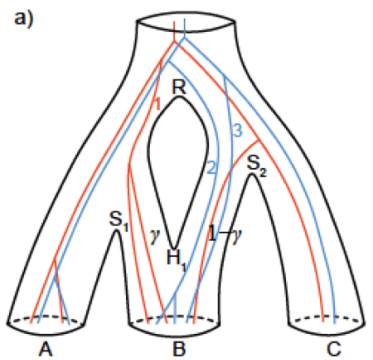
1. Inférence d'arbres d'espèces + arbres résumés en réseaux phylogénétiques

SNAPP (Bryant et al. 2012, MBE) + SplitsTree



2. Inférence directe de réseaux

SNAPPNet = extension de SNAPP aux réseaux

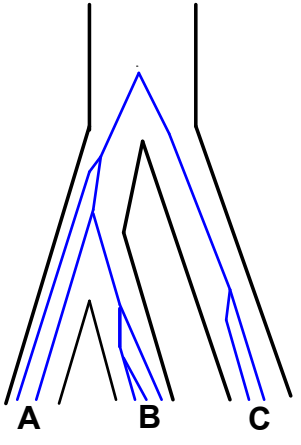


Plan

Logiciel SNAPP pour l'inférence Bayésienne d'arbres (Bryant et al. 2012, MBE)

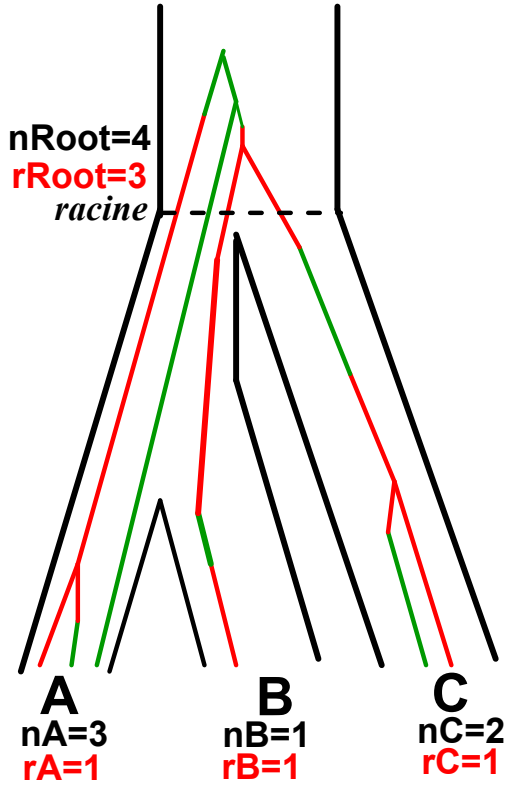
- Marqueurs bialléliques (SNPs) indépendants sachant l'arbre d'espèces

- Modélisation de l'arbre de locus (backward)
  - Processus de coalescence évoluant à l'intérieur d'un arbre d'espèces (MultiSpecies Coalescent)
  - Processus autorisant la discordance entre arbres de locus et arbres d'espèces (tri de lignées incomplet)



Les mutations interviennent au cours du temps

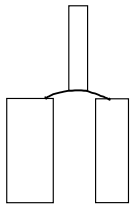
- Modélisation des séquences (forward)
  - mutation (rouge $\leftrightarrow$ vert) : modèle markovien évoluant le long des branches de l'arbre de locus
  - u : taux de mutation rouge $\rightarrow$ vert
  - v : taux de mutation vert $\rightarrow$ rouge



- V.a. : rRoot, nRoot, rA, rB, rC
- pas d'aléa dans nA, nB, nC
- Data=(rA, rB, rC)
- Vraisemblance : $\mathbb{P}(Data \mid S)$

La statistique Bayésienne dans SNAPP

- S : arbre d'espèces (topologie, longueurs de branches, tailles de populations)
- X_i : alignements pour le locus i
- G_i : arbre de locus pour le locus i
- m loci



Par la théorème de Bayes

$$\mathbb{P}(S|X_1,\ldots,X_m) \propto \left(\prod_{i=1}^{m}\int_{\psi}\mathbb{P}(X_i|G_i)\mathbb{P}(G_i|S)dG_i\right)P(S)$$

$$\propto \mathbb{P}\left(Data \mid S\right)P(S)$$

SNAPP intègre sur tous les arbres de locus

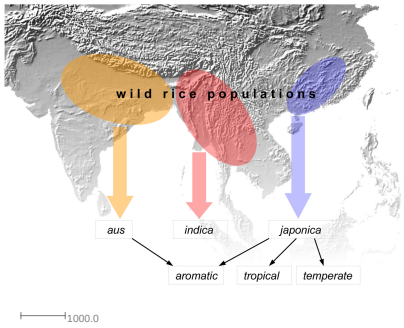
Calcul de la *prior* P(S) par le processus de naissances

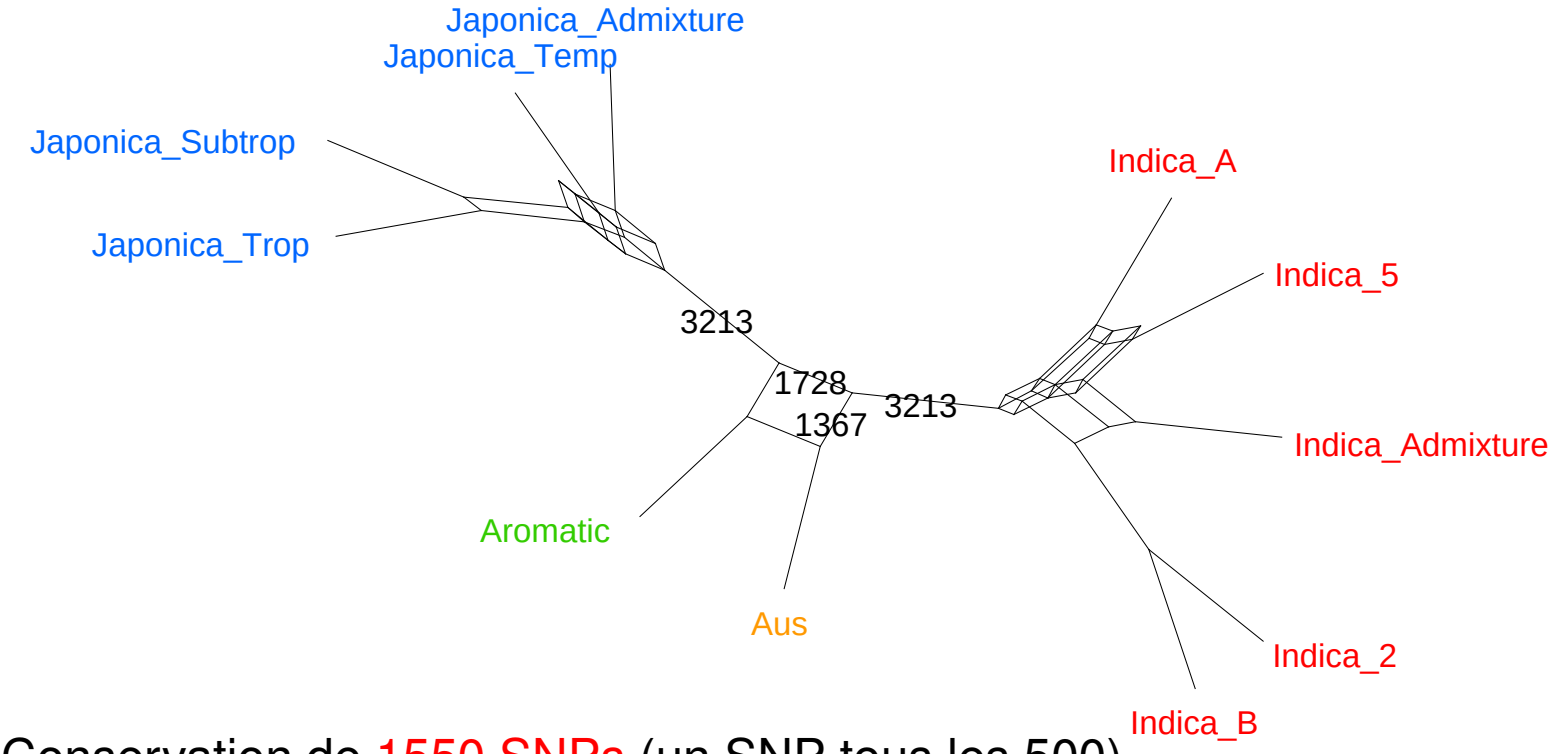
$\Rightarrow$ Markov Chain Monte Carlo (MCMC) afin d'estimer la distribution à posteriori de $\mathbb{P}(S|X_1,\ldots,X_m)$

Implémenté dans BEAST

Plan

Chromosome 6 (données J. Santos, J-C. Glaszmann)



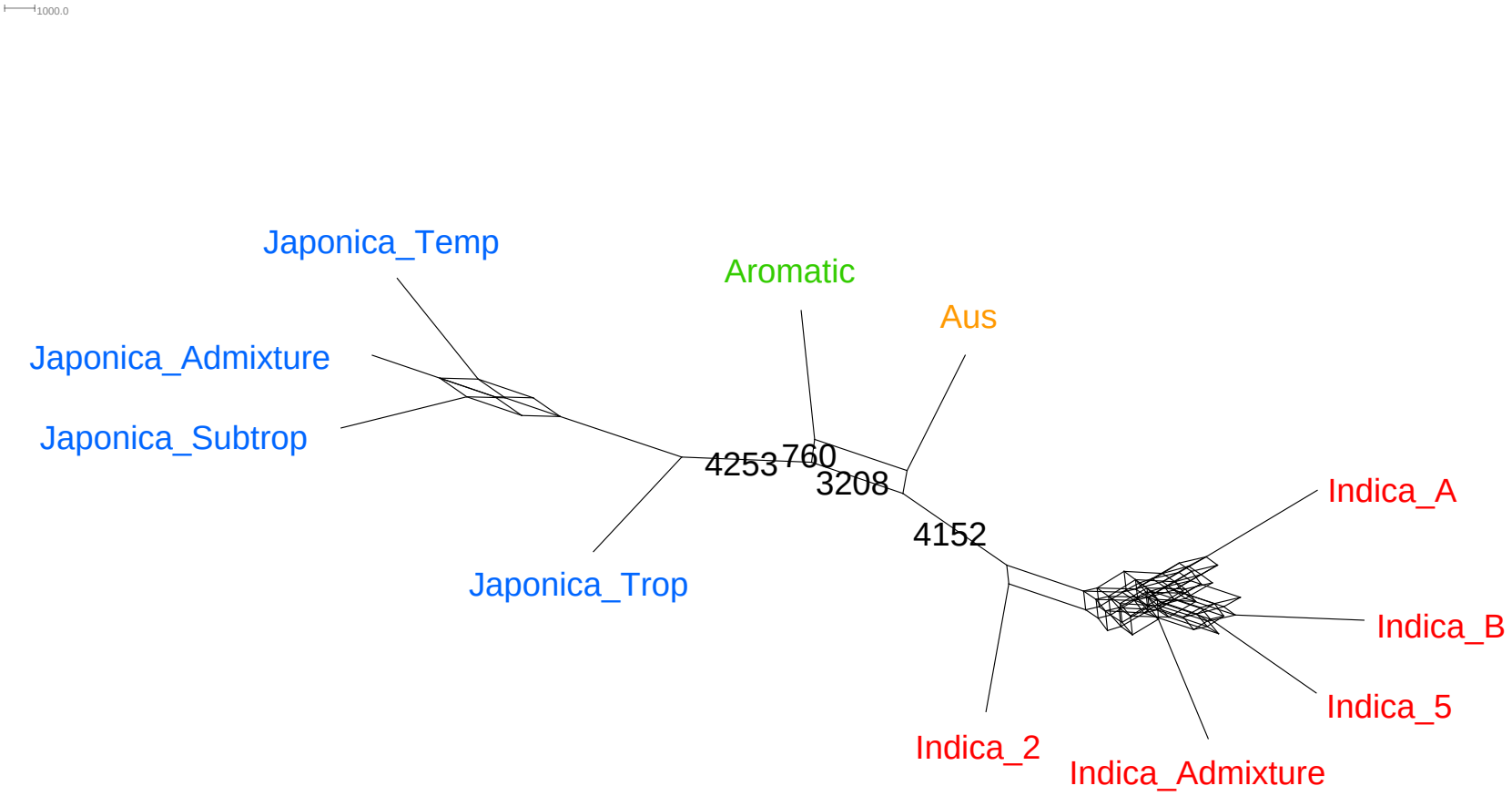


Conservation de 1550 SNPs (un SNP tous les 500)

Chromosome 10 (données J. Santos, J-C. Glaszmann)

Conservation de 1089 SNPs (un SNP tous les 500)

- JDD2 (1er SNP= 50ème SNP du chromosome 10)

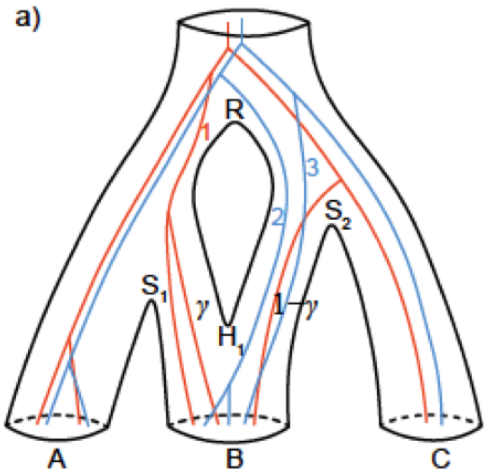


Plan

Simulateur basé sur un réseau (Genome Harvest)

SNAPPSimNet construit sur la base du simulateur SNAPPSim de Bryant et al. (2012)

- Génération d'arbres de locus évoluant à l'intérieur d'un réseau selon un processus de coalescence



- Snapp est fortement attiré par un scénario sous-jacent au réseau

Plan

Piste 2 : une méthode Bayésienne directe d'inférence de réseaux

- N : réseau phylogénétique (topologie, longueurs de branches, tailles de populations)
- X_i : alignements pour le locus i
- G_i : arbre de locus pour le locus i
- m loci

Par la théorème de Bayes

$$\mathbb{P}(N|X_1,\ldots,X_m) \propto \left(\prod_{i=1}^{m}\int_{\psi}\mathbb{P}(X_i|G_i)\mathbb{P}(G_i|S)dG_i\right)P(N)$$
$$\propto \mathbb{P}\left(Data \mid N\right)P(N)$$

SNAPPNet intègre sur tous les arbres de locus (extension de SNAPP, Bryant et al. MBE 2012), à l'aide d'un nouvel algorithme de parcours du réseau

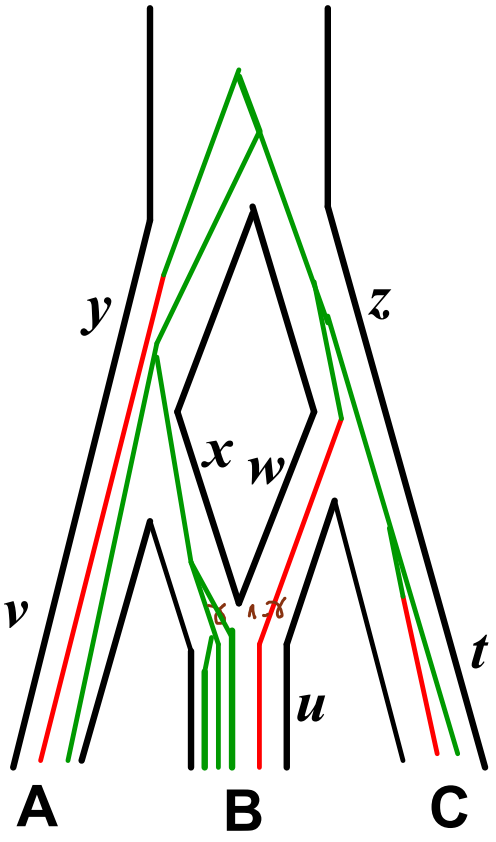
Calcul de la *prior* P(N) par le processus de naissances hybridation

$\Rightarrow$ Markov Chain Monte Carlo (MCMC) afin d'estimer la distribution à posteriori de $\mathbb{P}(N|X_1,\ldots,X_m)$

Implémenté dans BEAST

Plan

Problème sous-jacent aux réseaux phylogénétiques



$Data_z$: proportion de rouge/vert dans les espèces sous la branche z

$Data_y$: proportion de rouge/vert dans les espèces sous la branche y

$Data_{zT}$ et $Data_{yT}$ ne sont plus indépendantes...

$Data_{zT}$ et $Data_{yT}$ comprennent les allèles rouges et verts de l'espèce hybride B !!!

On ne peut plus effectuer le produit des probabilités

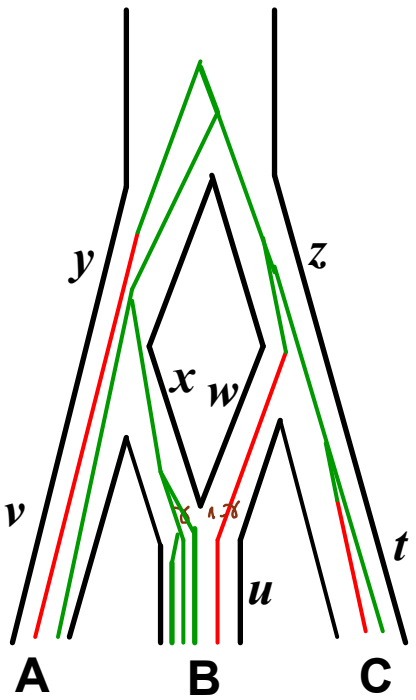
$\mathbb{P}(Data_{zT}) \times \mathbb{P}(Data_{yT})$!!!

Calcul de la vraisemblance dans un réseau

$$
\begin{aligned}
&\mathbb{P}(Data) \\
=&\sum_i \sum_j \mathbb{P}(Data \mid n_{root} = i, r_{root} = j)\, \mathbb{P}(r_{root} = j \mid n_{root} = i) \\
&\mathbb{P}(n_{root} = i) \\
=&\sum_i \sum_j \sum_{i'} \sum_{j'} \mathbb{P}\big(Data_{zT}\, Data_{yT} \mid n_{yT} = i', n_{zT} = i - i', r_{yT} = j', \\
&\quad r_{zT} = j - j'\big)\, \mathbb{P}\big(r_{yT} = j',\ r_{zT} = j - j' \mid n_{yT} = i', n_{zT} = i - i', r_{root} = j\big) \\
&\quad \mathbb{P}\big(n_{yT} = i', n_{zT} = i - i' \mid n_{root} = i\big)\, \mathbb{P}(r_{root} = j \mid n_{root} = i)\, \mathbb{P}(n_{root} = i)
\end{aligned}
$$

On ne peut plus effectuer le produit des probabilités
$\mathbb{P}(Data_{zT}) \times \mathbb{P}(Data_{yT})$!!!

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

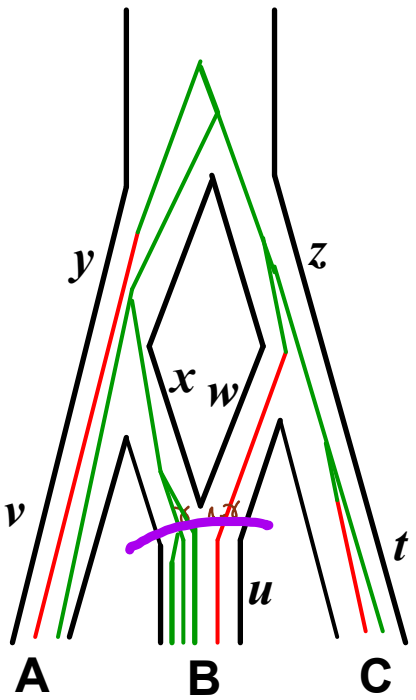
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

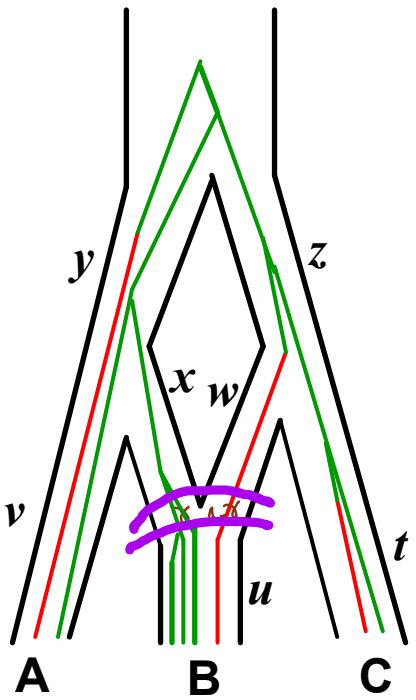
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

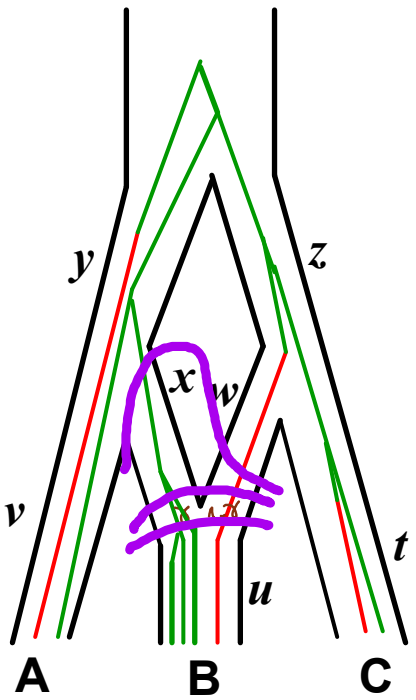
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

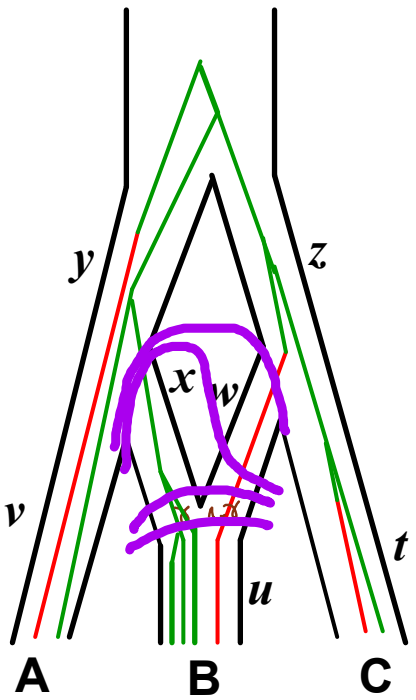
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

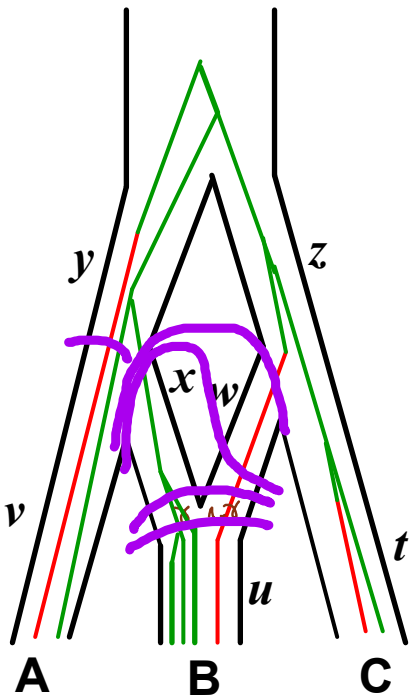
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

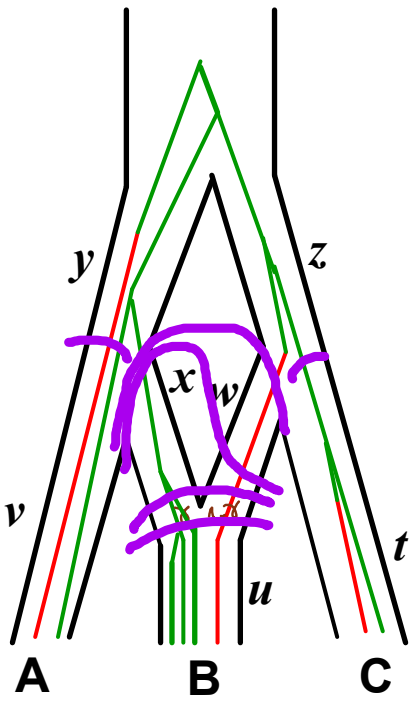
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

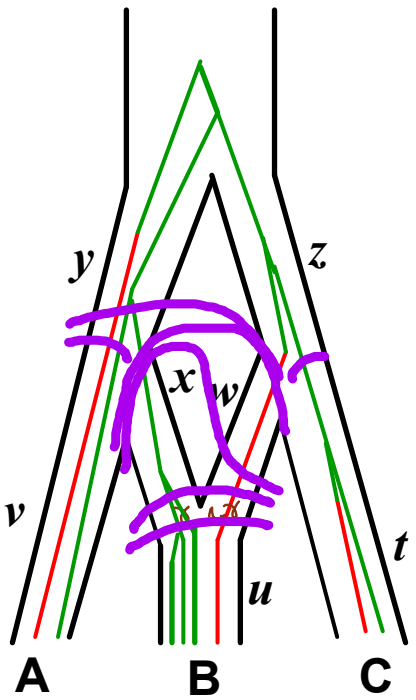
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

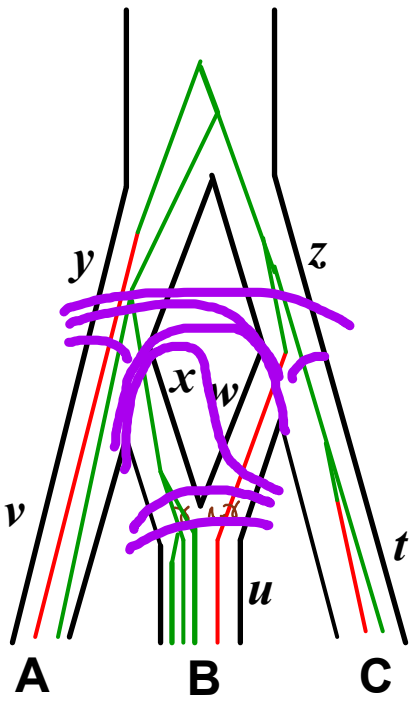
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

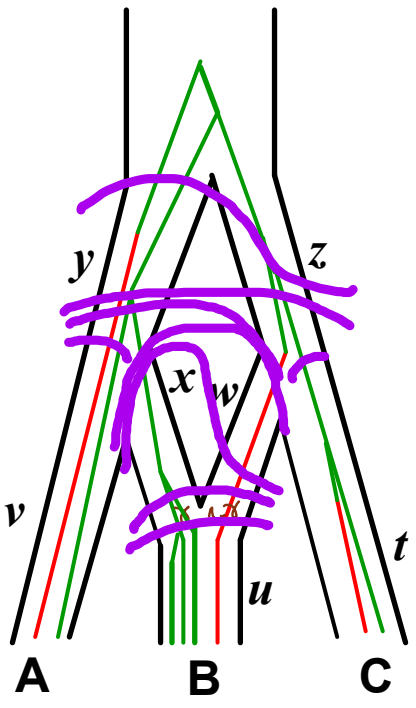
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

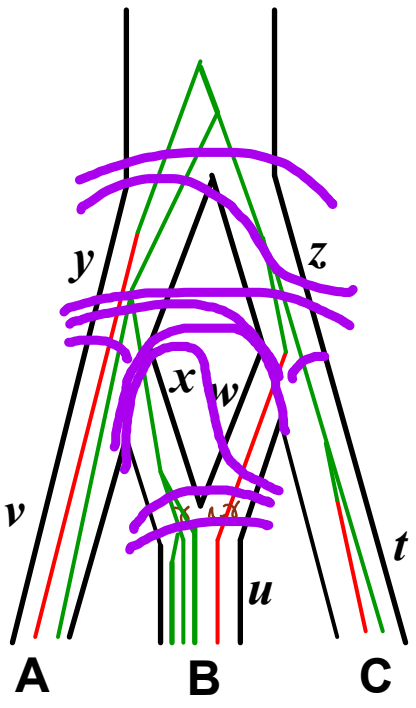
(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Notre algorithme : calcul des lois jointes



Quantités calculées successivement

(1) $\mathbb{P}\left(Data_{uT} \mid n_{uT}, r_{uT}\right)$

(2) $\mathbb{P}\left(Data_{xB}\, Data_{wB} \mid n_{xB}, r_{xB}, n_{wB}, r_{wB}\right)$

(3) $\mathbb{P}\left(Data_{xT}\, Data_{wB} \mid n_{xT}, r_{xT}, n_{wB}, r_{wB}\right)$

(4) $\mathbb{P}\left(Data_{xT}\, Data_{wT} \mid n_{xT}, r_{xT}, n_{wT}, r_{wT}\right)$

(5) $\mathbb{P}\left(Data_{vT} \mid n_{vT}, r_{vT}\right)$

(6) $\mathbb{P}\left(Data_{tT} \mid n_{tT}, r_{tT}\right)$

(7) $\mathbb{P}\left(Data_{yB}\, Data_{wT} \mid n_{yB}, r_{yB}, n_{wT}, r_{wT}\right)$

(8) $\mathbb{P}\left(Data_{yB}\, Data_{zB} \mid n_{yB}, r_{yB}, n_{zB}, r_{zB}\right)$

(9) $\mathbb{P}\left(Data_{yT}\, Data_{zB} \mid n_{yT}, r_{yT}, n_{zB}, r_{zB}\right)$

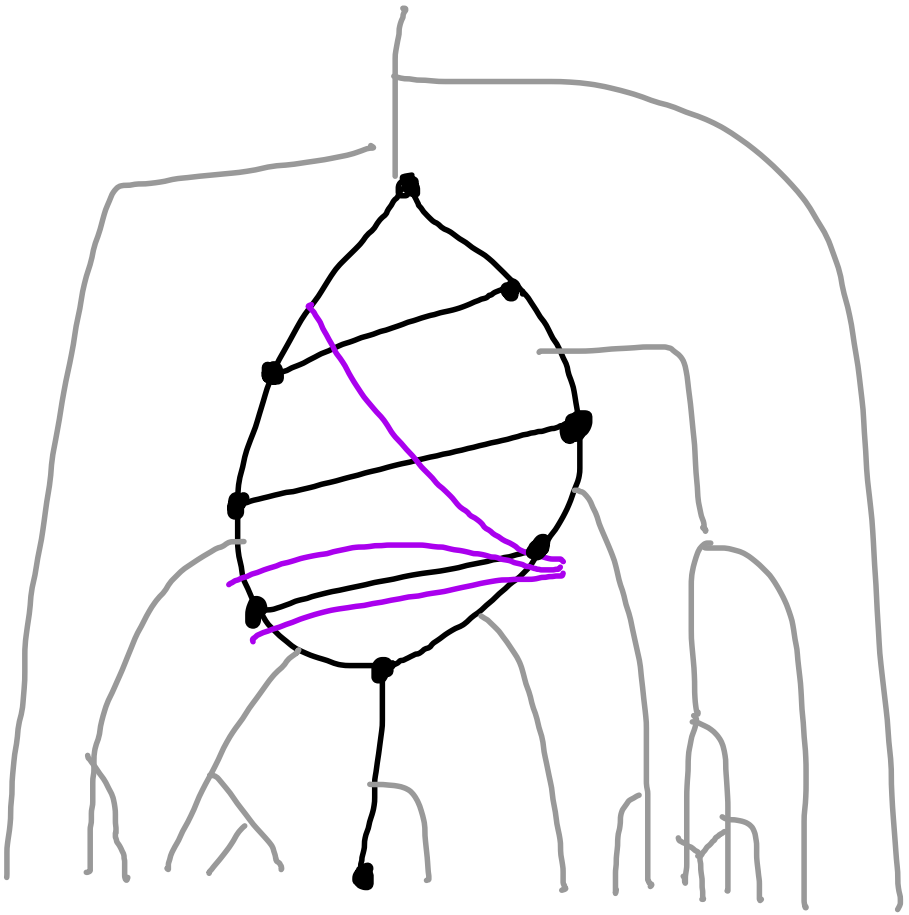
(10) $\mathbb{P}\left(Data_{yT}\, Data_{zT} \mid n_{yT}, r_{yT}, n_{zT}, r_{zT}\right)$

(11) $\mathbb{P}\left(Data \mid n_{root}, r_{root}\right)$

Nous cherchons à minimiser le nombre d'arêtes à considérer simultanément dans nos calculs de probabilités

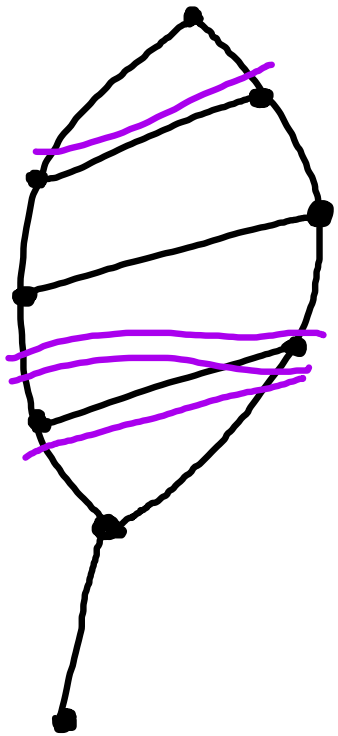
Un parcours à éviter

Un maximum de 5 arêtes



Un parcours intéressant

Un maximum de 3 arêtes



Plan

A propos du prior sur le réseau (Zhang et al., MBE 2017)

Rappel au niveau du posterior :

$$\mathbb{P}(N|X_1, \ldots, X_m) \propto \mathbb{P}\left(Data \mid N\right) P(N)$$

Le prior est $P(N)$.
Le processus de naissance hybridation dépend de paramètres ν et λ.

Des lois sont imposées sur ν et λ : on parle d'hyper prior.

$$\mathbb{P}(N, X, Y) = \mathbb{P}(N \mid X, Y)P(X)P(Y)$$

- X : v.a. pour la valeur $\lambda - \nu$
- la loi choisie pour X est une loi exponentielle (Zhang et al., MBE 2017)
- Y : v.a. pour la valeur $\frac{\nu}{\lambda}$
- la loi choisie pour Y est une loi Beta (Zhang et al., MBE 2017)

A propos du xml de notre logiciel SNAPPNet (Add On pour Beast)

<distribution id="networkPrior"
spec="speciesnetwork.BirthHybridizationModel"
network="@network :species" netDiversification="@netDivRate :species"
turnOver="@turnOverRate :species"/>
<prior id="netDivPrior" name="distribution" x="@netDivRate :species">
<Exponential id="exponential.01" name="distr" mean="10.0"/>
</prior>
<prior id="turnOverPrior" name="distribution" x="@turnOverRate :species">
<Beta id="betadistr.01" name="distr" alpha="1.0" beta="1.0"/>
</prior>

A plus long terme :
BEAUti — Bayesian Evolutionary Analysis Utility.
This program is used to import data, design the analysis, and generate the BEAST control file.

Plan

A propos des opérateurs implémentés

16 opérateurs pour l'échantillonnage par MCMC (Bryant et al, MBE 2012 ; Zhang et al., MBE 2017)

Opérateurs de changement topologique

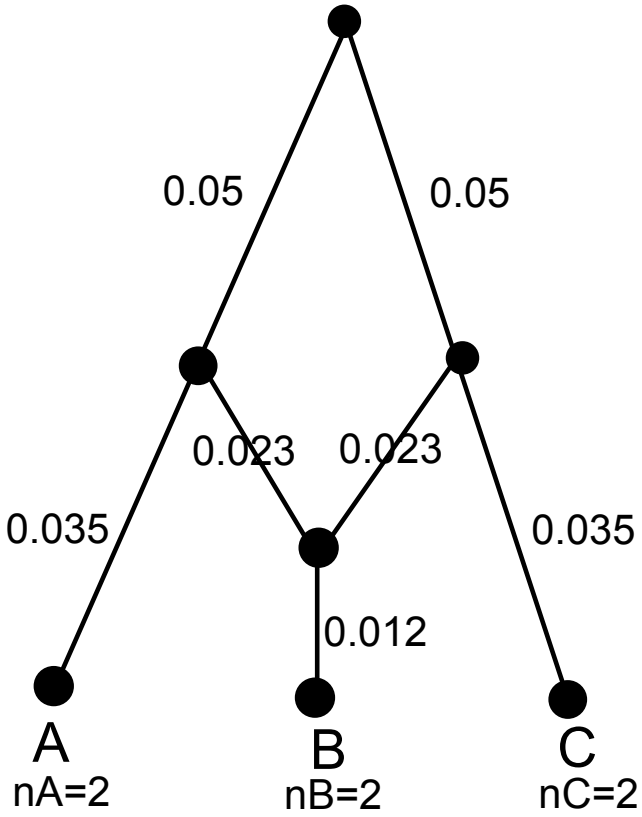
- addReticulation : ajout d'un noeud de réticulation
- deleteReticulation : suppression d'un noeud de réticulation
- flipReticulation : inversion de l'orientation d'une branche de réticulation
- relocateBranch

Autres opérateurs, comme par exemple

- MutationMover : changement des valeurs des taux de mutation u (rouge $\rightarrow$ vert) et v (vert $\rightarrow$ rouge), sous la contrainte $\frac{2uv}{u+v} = 1$
- ChangeTheta : changement de la taille de population θ liée à une branche
- ChangeAllTheta : changement de toutes les tailles de population θ
- turnOverScale : changement de la valeur du paramètre $\frac{\nu}{\lambda}$ lié au processus de naissance hybridation
  (ν taux d'hybridation, λ taux de spéciation)
- divrRateScale : changement de la valeur du paramètre λ - ν lié au processus de naissance hybridation

Plan

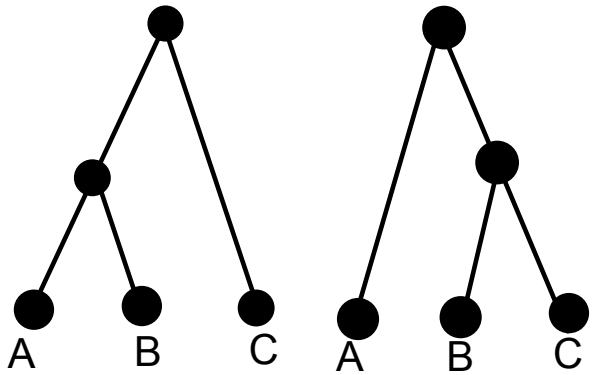
Un exemple sur données simulées



- Longueurs de branches en nombre de mutations par site
- nA=2, nB=2, nC=2
- 1 000 sites ou 10 000 sites
- Tailles de population θ égales à 0.005 ou 0.05
- T : temps de coalescence entre 2 lignées (en mutations par site)
  - si $\theta = 0.005$, alors $\mathbb{E}(T) = 0.005/2 = 0.0025$
  - si $\theta = 0.05$, alors $\mathbb{E}(T) = 0.005/2 = 0.025$

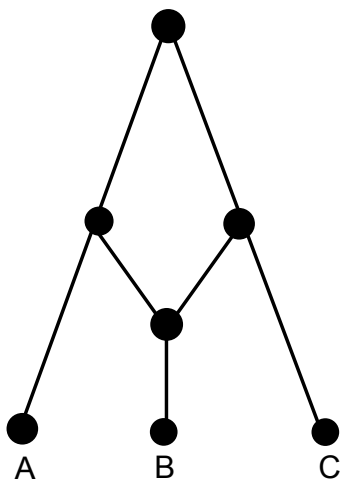
Résultats obtenus par MCMC

- 1 000 sites, $\theta = 0.005$



51.94% 48.06%

- 10 000 sites, $\theta = 0.005$
- 1 000 sites, $\theta = 0.05$
- 10 000 sites, $\theta = 0.05$



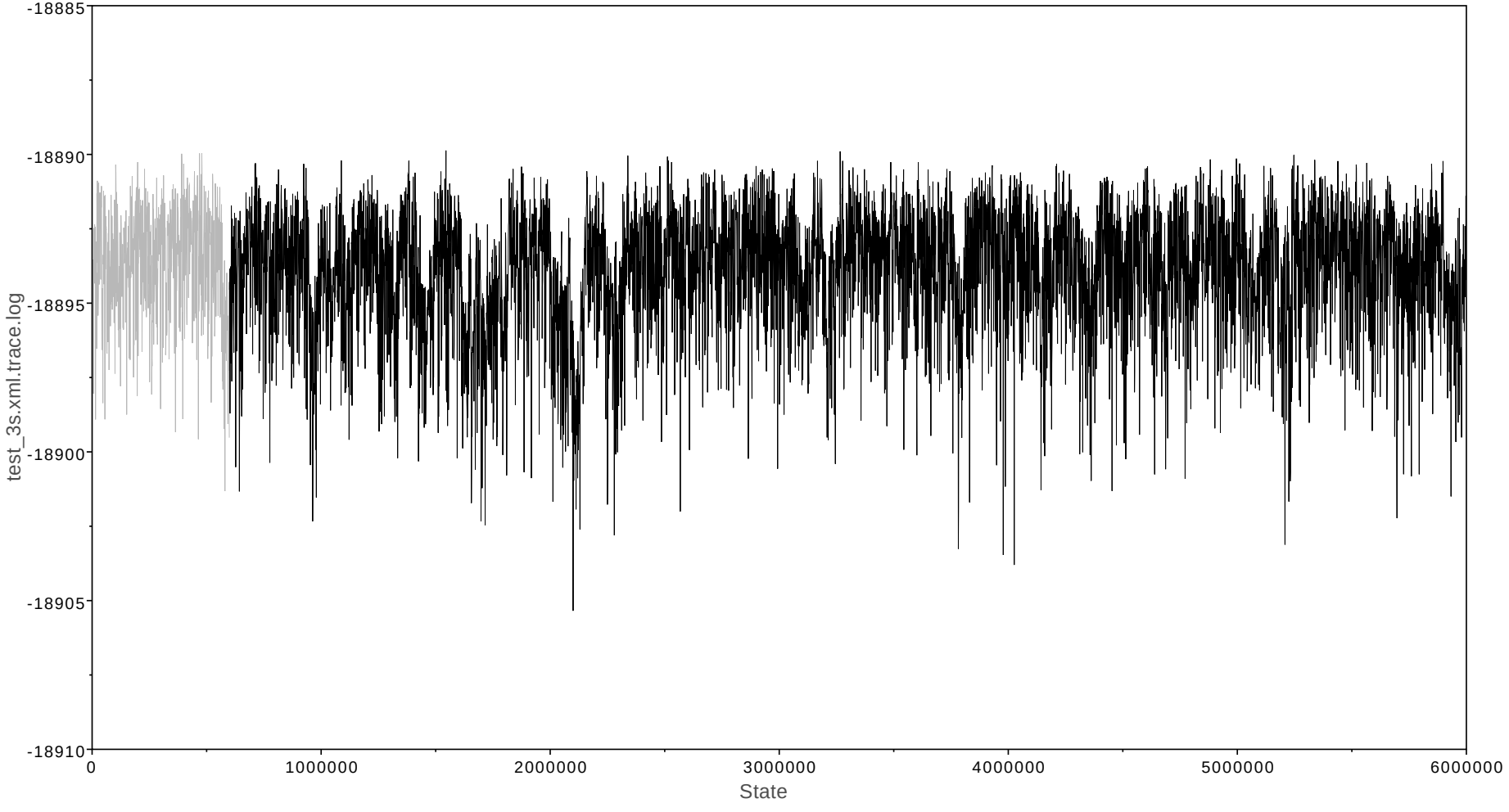
100%

Avec une taille de population plus importante,
on a besoin de moins de sites pour retrouver le réseau ! ! !

Distribution a posteriori échantillonnée
Cas 10 000 sites, $\theta = 0.05$

Analyse avec le logiciel Tracer

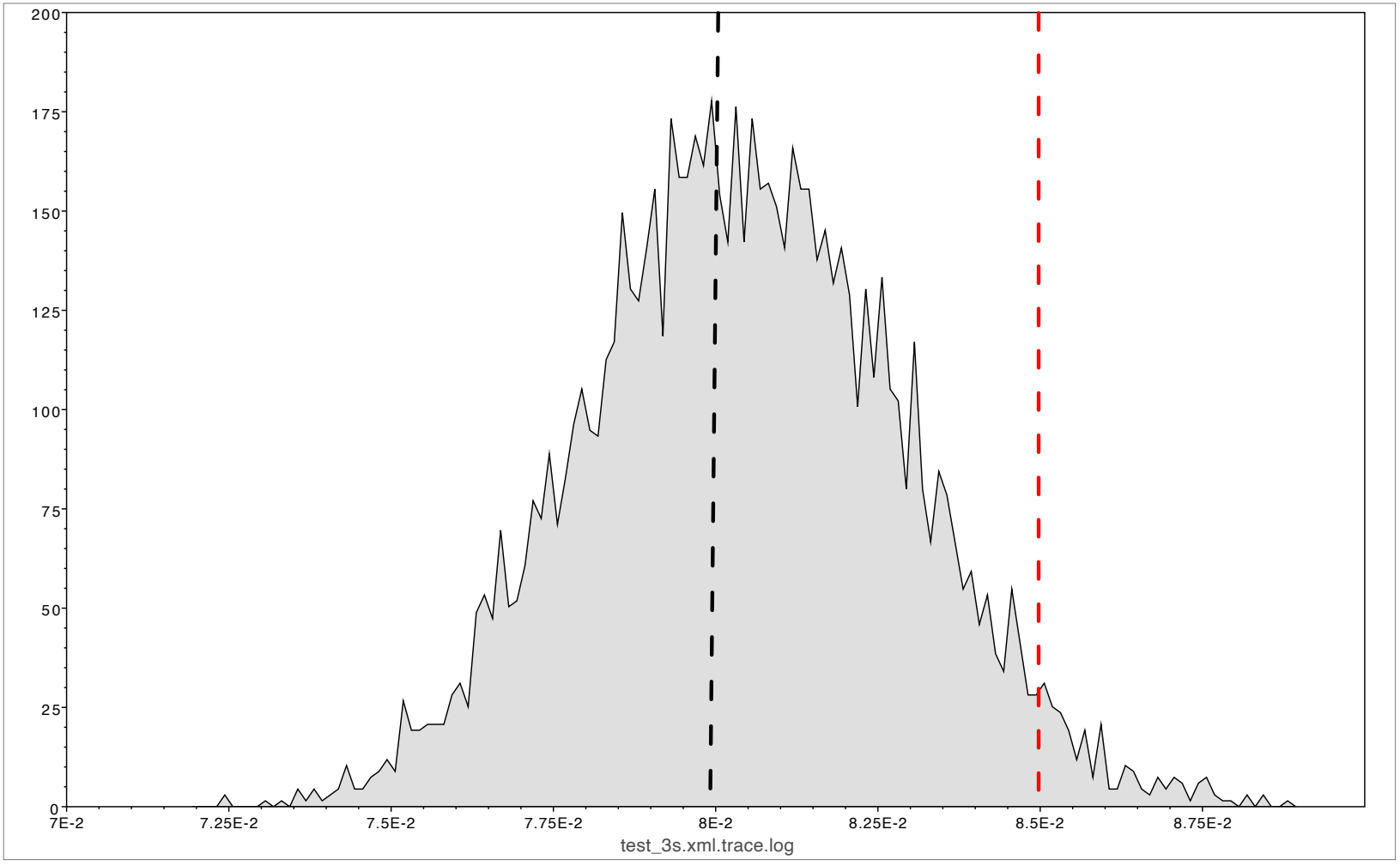
Effective Sample Size (ESS) = 413



A propos de la hauteur estimée de notre réseau
Cas 10 000 sites, $\theta = 0.05$

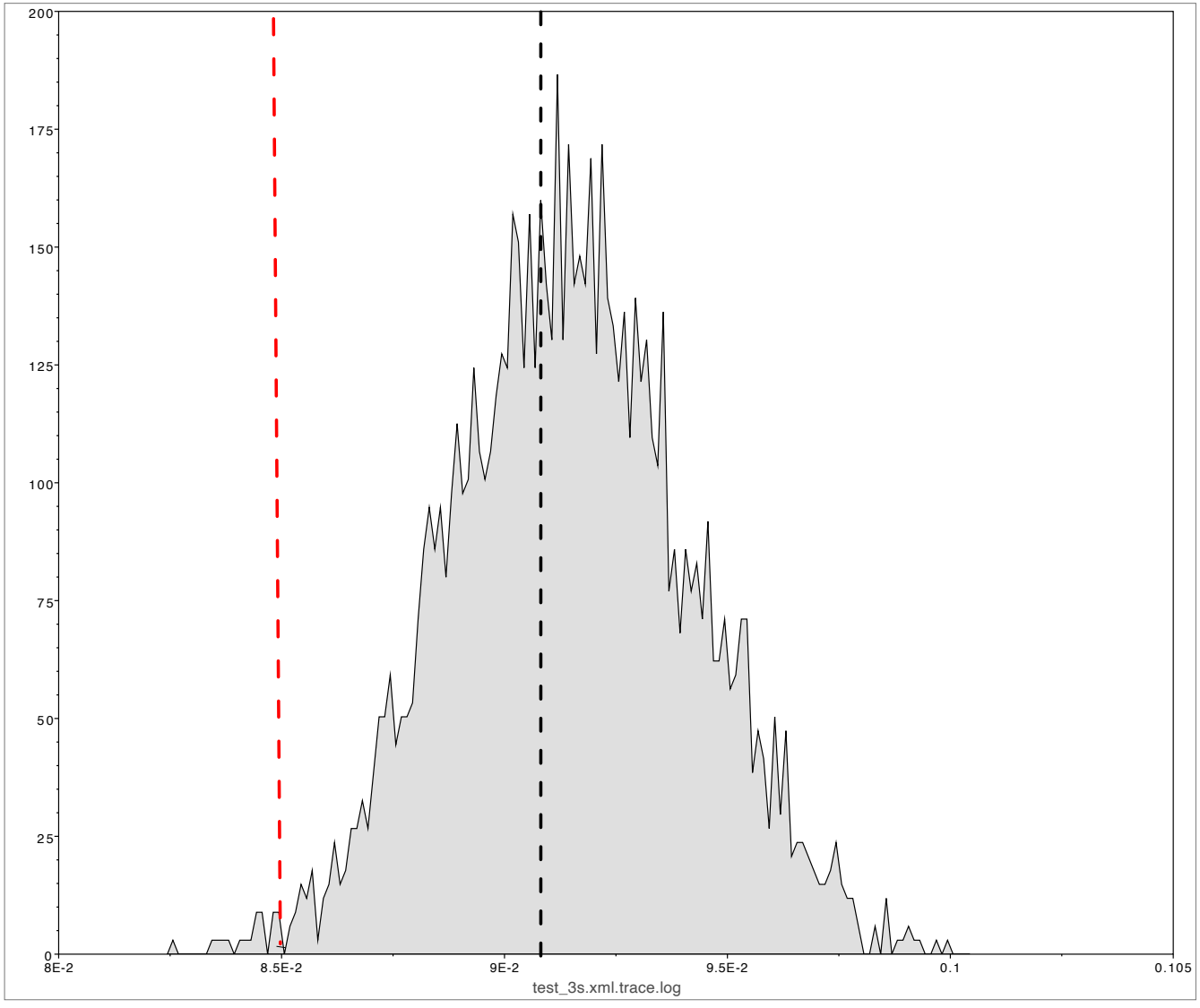
Estimateur considéré par Tracer : Moyenne a posteriori

$\widehat{\text{height}} = \mathbb{E}(\text{height} \mid \text{Data}) = 0.08044$ Vraie valeur = 0.085



Cas 10 000 sites, $\theta = 0.005$

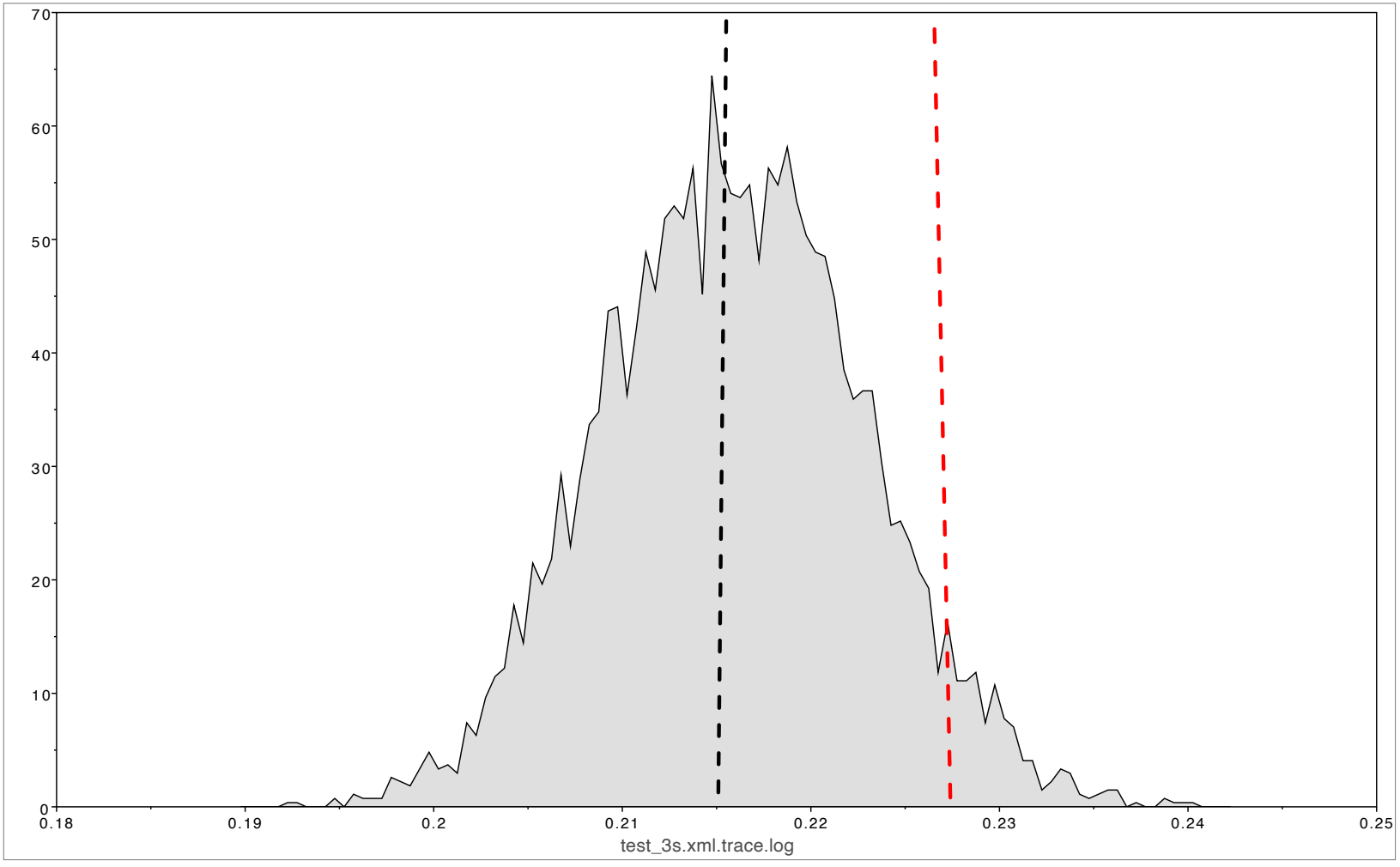
$\hat{\text{height}} = \mathbb{E}(\text{height} \mid \text{Data}) = 0.0915$ Vraie valeur = 0.085



A propos de la longueur estimée de notre réseau
Cas 10 000 sites, $\theta = 0.05$

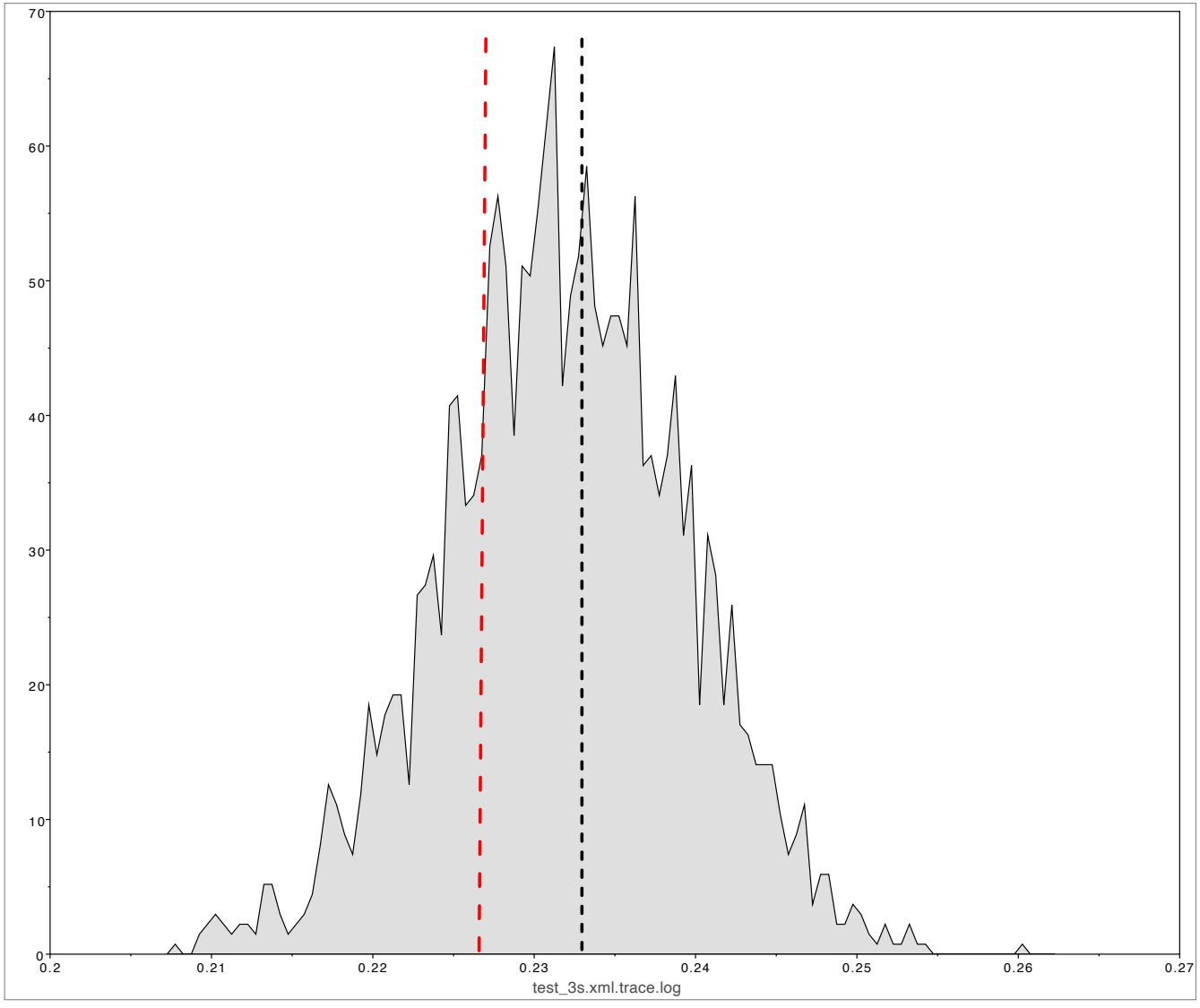
Estimateur considéré par Tracer : Moyenne a posteriori

$\hat{\text{length}} = \mathbb{E}(\text{length} \mid \text{Data}) = 0.216$ Vraie valeur = 0.228

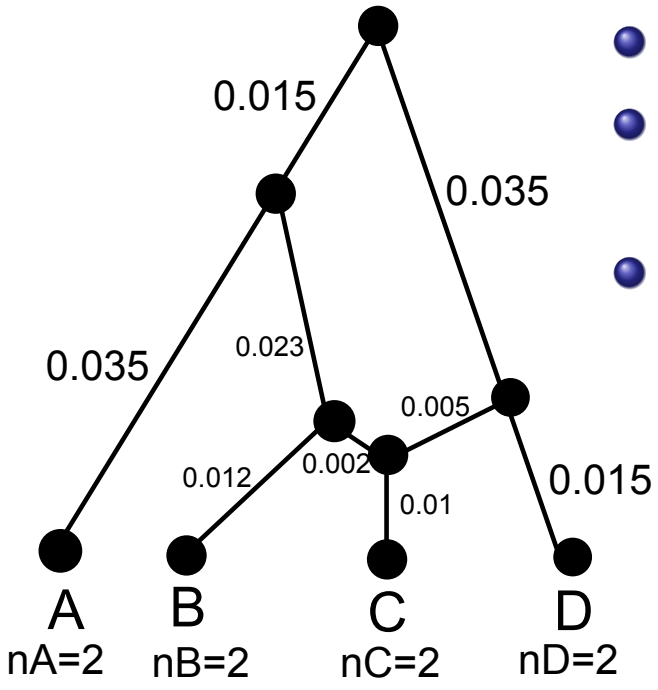


Cas 10 000 sites, $\theta = 0.005$

$\hat{\text{length}} = \mathbb{E}(\text{length} \mid \text{Data}) = 0.232$ Vraie valeur = 0.228



Un réseau un peu plus compliqué



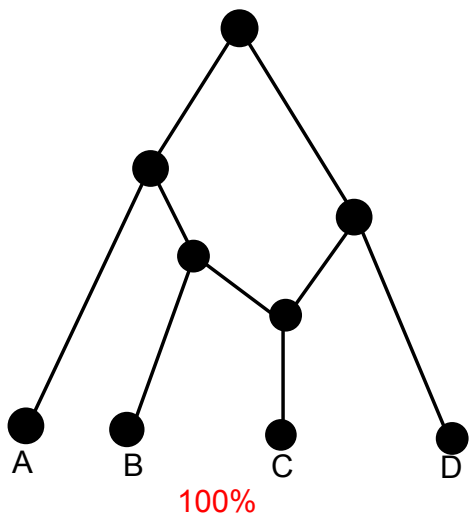
- Longueurs de branches en nombre de mutations par site
- nA=2, nB=2, nC=2, nD=2
- 1 000 sites ou 10 000 sites
- Tailles de population θ égales à 0.005 ou 0.05
- T : temps de coalescence entre 2 lignées (en mutations par site)
  - si $\theta = 0.005$, alors $\mathbb{E}(T) = 0.005/2 = 0.0025$
  - si $\theta = 0.05$, alors $\mathbb{E}(T) = 0.005/2 = 0.025$

Résultats obtenus par MCMC

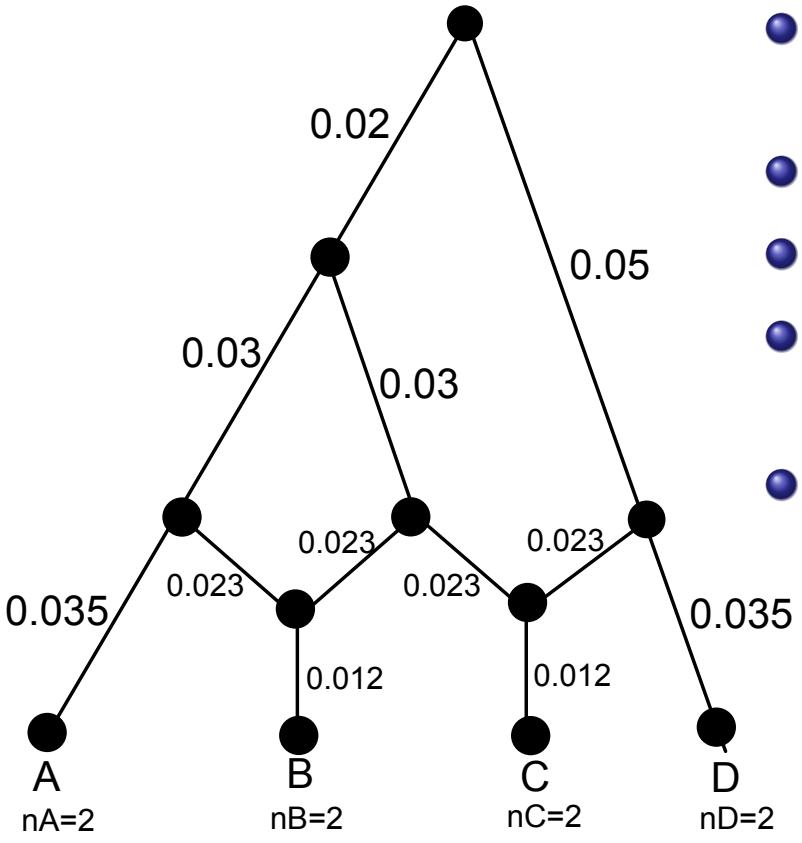
- 1 000 sites, $\theta = 0.005$
- 1 000 sites, $\theta = 0.05$

Arbres non nécessairement inclus dans le réseau

- 10 000 sites, $\theta = 0.005$
- 10 000 sites, $\theta = 0.05$



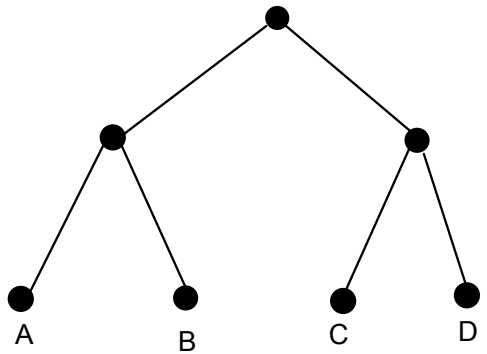
Un réseau avec 2 réticulations



- Longueurs de branches en nombre de mutations par site
- nA=2, nB=2, nC=2, nD=2
- 1 000 sites ou 10 000 sites
- Tailles de population θ égales à 0.005 ou 0.05
- T : temps de coalescence entre 2 lignées (en mutations par site)
  - si $\theta = 0.005$, alors $\mathbb{E}(T) = 0.005/2 = 0.0025$
  - si $\theta = 0.05$, alors $\mathbb{E}(T) = 0.005/2 = 0.025$

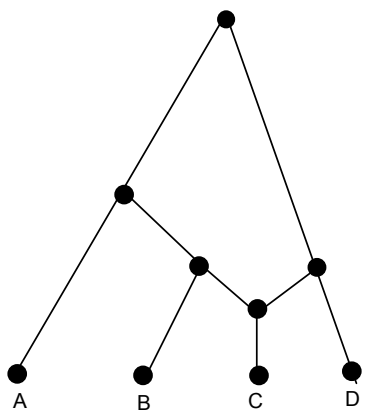
Résultats obtenus par MCMC

- 1 000 sites

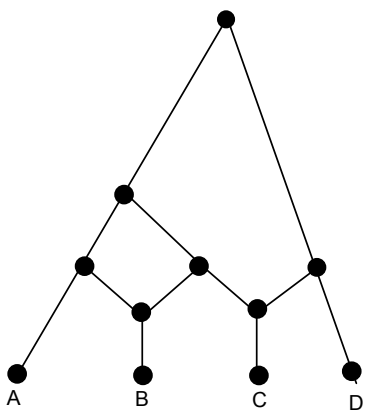


98.44% pour theta=0.05
99.90% pour theta=0.005

- 10 000 sites



100% pour theta=0.005



98.99% pour theta=0.05

Avec une taille de population importante,
et un grand nombre de sites, on arrive à retrouver le réseau a 2 reticulations ! !

Plan

Conclusion

- Exploration de la piste SNAPP+SplitsTree depuis données des 3000 génomes de riz
- Nouvelle méthode Bayésienne de reconstruction de réseaux phylogénétiques
- L'inférence de réseau est un sujet compétitif : Tanja Stadler (ETH Zurich), Luay Nakhleh (Rice University, USA). Notre approche s'avère plus performante sur des réseaux aux nombreuses hybridations
- Il nous reste à publier
- En cours de test sur données réelles (jeux de Wang et al, 2017, riz sauvages et cultivés)
- Jusqu'alors travail sur le riz $\rightarrow$ autre plante d'intérêt ? Genome Harvest : Banane, Citrus, Caféier, Tomate, Canne à sucre ...
- Il est possible de considérer en entrée de la méthode un squelette avec évènements évolutifs connus (e.g. hybridation)